AFIT/GAE/ENY/91D-13

1

INVESTIGATION OF A FLUX-DIFFERENCE-SPLITTING NUMERICAL METHOD IN SUPERSONIC NOZZLES

THESIS

Danny P. Allen, Captain, USAF

AFIT/GAE/ENY/91D-13

Approved for public release; distribution unlimited

AFIT/GAE/ENY/91D-13

INVESTIGATION OF A FLUX-DIFFERENCE-SPLITTING NUMERICAL METHOD IN SUPERSONIC NOZZLES

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Masters of Science in Aeronautical Engineering

Danny P. Allen, M.S.
Captain, USAF

December 1991

Approved for public release; distribution unlimited

Acknowledgements

During this research effort, I have had a great deal of help from my advisor, Capt John H. Doty. Through his many hours of assistance and advice I have been able to complete this work for which I am deeply indebted. Additionally, I would like to thank him for allowing me to borrow some of the figures in his research dissertation. I also would like to thank my wife Deborah for her patience and understanding during my time at the Air Force Institute of Technology.

Table of Contents

List of Figures

List of Symbols

a	speed of sound
$d\boldsymbol{E}$	difference in E flux vector
$d\boldsymbol{F}$	difference in F flux vector
e	specific internal energy
$\boldsymbol{E}$	E flux vector
$\boldsymbol{F}$	F flux vector
M	Mach number
P	static pressure
u	axial component of velocity
v	radial component of velocity
V	magnitude of velocity
x	axial direction in the physical plane
y	radial direction in the physical plane

Greek Symbols

α	Mach angle
β	difference between shock wave angle and flow deflection angle
γ	specific heat ratio
δ	flow deflection angle
Δ_i	finite difference operator in the i direction
Δ_j	flux difference operator in the j direction
ε	shock wave angle
ζ	transformed axial coordinate

η	transformed radial coordinate
θ	flow angle
Λ	characteristic slope in transformed plane
ν	Prandtl-Meyer angle
ρ	density
σ	streamline slope

Subscripts and Superscripts

i	axial direction index in physical plane
j	radial direction index in physical plane
j-1/2	Riemann location between nodes j and j-1
j-3/2	Riemann location between nodes j-1 and j-2
j+1/2	Riemann location between nodes j and j+1
j+3/2	Riemann location between nodes j+1 and j+2
x,y	partial derivative with respect to that variable
wave 1,2, or 3	Riemann waves
0, 2, 4, 6	Riemann regions
+	positively biased split flux difference
-	negatively biased split flux difference

AFIT/GAE/ENY/91D-13

Abstract

This research investigated the effects of adding a second-order flux-difference-splitting (FDS) correction term to an existing computer code that is based on a first-order FDS algorithm. It was determined that the second-order algorithm did improve the accuracy of the code for a source flow analysis, but second-order behavior could not be confirmed by the error convergence patterns. It was also discovered that, when tested across an oblique shock wave, the second-order correction terms had minimal influence on the accuracy and shock capturing ability of the first-order accurate FDS method.

INVESTIGATION OF FLUX-DIFFERENCE-SPLITTING NUMERICAL METHOD IN SUPERSONIC NOZZLES

I. Introduction

Purpose

The purpose of this research is to add a second-order accuracy to a newly developed first-order accurate computer code that predicts the inviscid, two-dimensional (2D) flow properties within and around the nozzles of hypersonic aerospace vehicles, such as the National Aerospace Plane (NASP). The code, which was developed by Doty (1), is based upon an innovative differencing technique, known as Flux-Difference-Splitting (FDS). The flux-difference-splitting numerical method was created by Enquist and Osher (5:45-75). Doty's code uses a first-order accurate upwind FDS method, operating on the steady, inviscid, planar form of the Euler Equations to model the flow field. Additionally, the fluid is modeled with a perfect gas assumption. This thesis adds second-order correction terms to the first-order accurate code and evaluates the results by comparing them against the existing first-order accurate solution and known exact solutions. The exact solutions used for this comparison are supersonic source flow and shock wave reflection.

Background

Over the last twenty years, computers have played an increasing role in the design of aircraft. Until recently, an aircraft design would become finalized, and then a scaled down model of the finished design would be built and tested in a wind tunnel. This approach proved to be very expensive, time consuming, and inefficient. Additionally, it lacked the flexibility needed to improve the design as contract or performance requirements changed, or to allow optimization of the design during the design phase. Now, with the advancement of supercomputers, engineers have the capability to mathematically model and simulate flights of the current unfinished design, making it possible to optimize the design before it is finalized.

According to Barthelemy (2:6-9), the NASP will require technology that is a quantum leap ahead of the technology that is used in today's aircraft and spacecraft, and will expend enormous amounts of government and contractor resources. A sketch of a typical NASP type vehicle can be seen in Figure 1. This type of vehicle will be powered by a supersonic combustion ramjet (SCRAMJET) and, theoretically, will be able to cruise at speeds of up to Mach 25.

The computer code was written as a "user friendly" tool that design engineers could utilize to predict flow parameters in a NASP type nozzle to enhance design optimization. These predicted flow parameters are shock wave and slip stream locations, pressures, temperatures, and velocities throughout the plane of the nozzle.

Figure 2 shows the general configuration of a NASP type nozzle. In Figure 2, region 1 is the combustor exit. Region 2 is the external air flow that passes under the engines. The exhaust flow, in region 3a, and the external air flow, in region 3b, are separated by a contact surface that originates from the engine cowl, as illustrated in the figure.

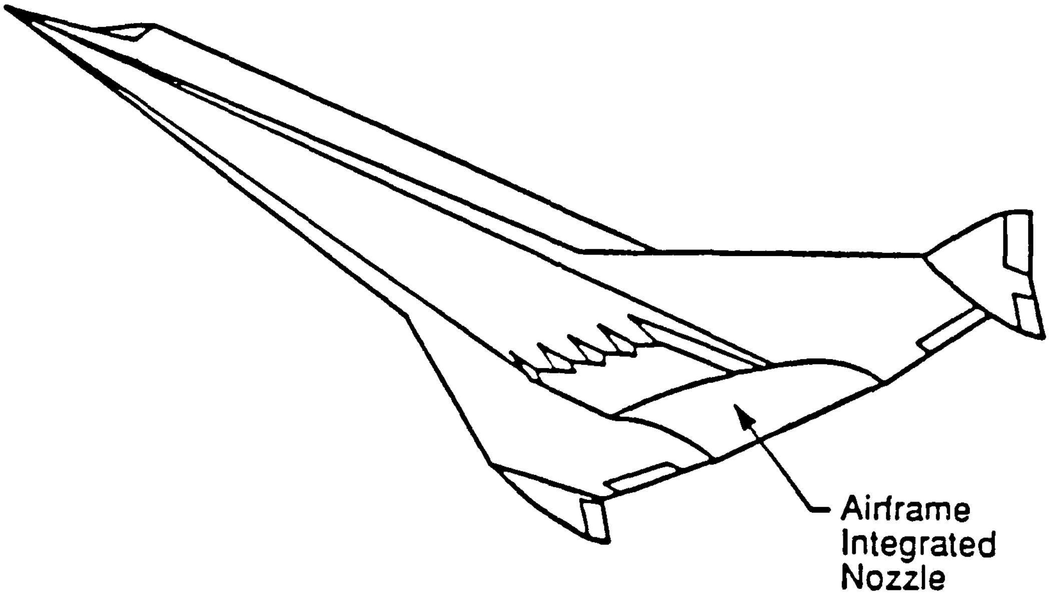


Figure 1. Typical hypersonic vehicle (3:4).

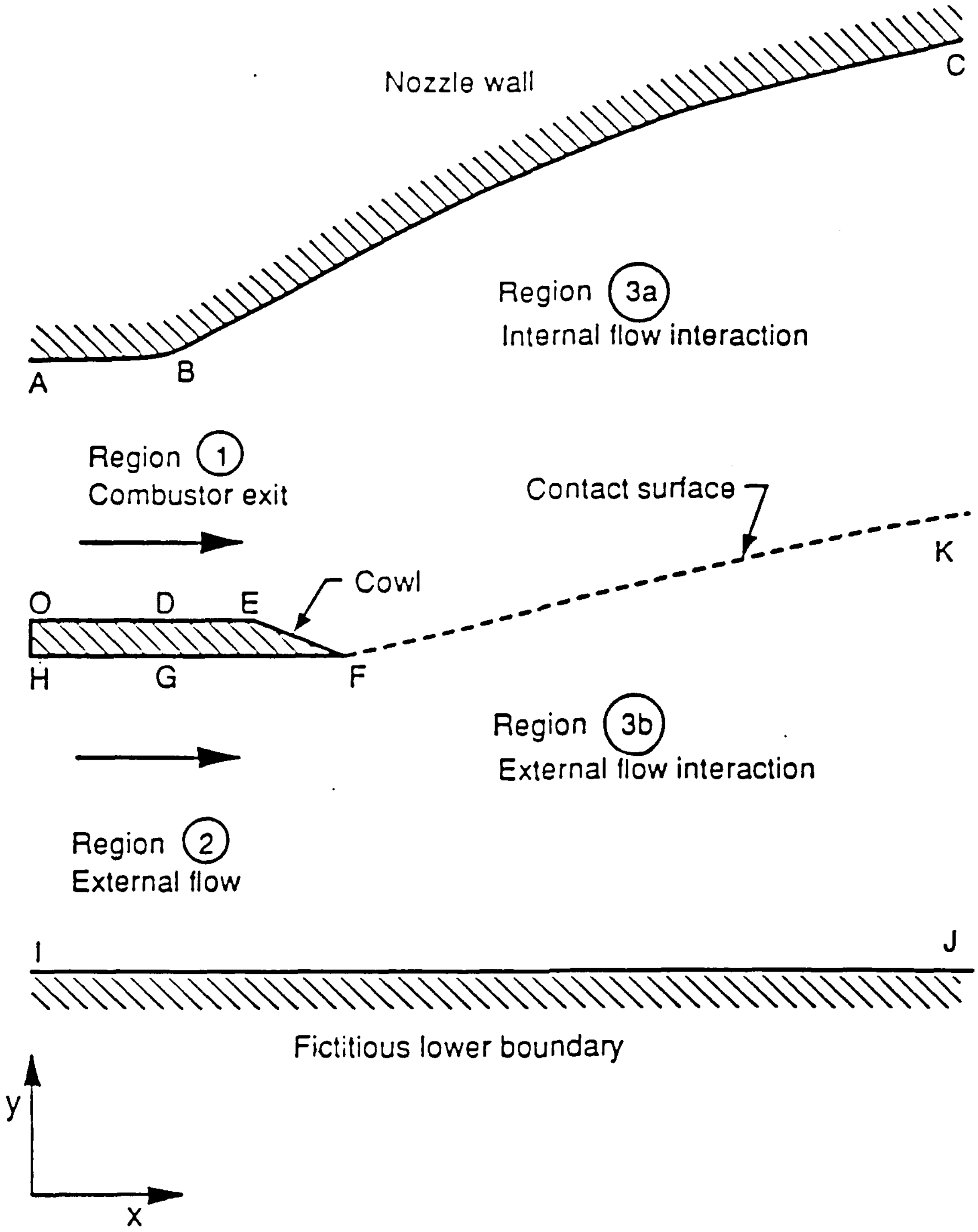


Figure 2. Expanded view of nozzle and cowl section (3:5).

II. Governing Equations and Coordinate Transformation

Governing Equations

According to Anderson et al. (1:235-236), for flows with sufficiently high Reynolds numbers the viscous and heat transfer effects are confined to a thin boundary layer near the wall. For a large ducted supersonic flow, such as flow in a NASP type nozzle, these effects can be neglected for preliminary analysis and design. With these two assumptions applied to the full Navier-Stokes equations, and neglecting body forces, the governing equations for this 2D, steady, compressible, inviscid, adiabatic fluid flow can be reduced to the inviscid Euler Equations:

$$\frac{\partial \boldsymbol{E}}{\partial x} + \frac{\partial \boldsymbol{F}}{\partial y} = 0 \qquad (1)$$

where

$$\boldsymbol{E} = \begin{bmatrix} \rho u \\ \rho u^2 + P \\ \rho u v \\ u(\rho e + P) \end{bmatrix} \qquad \boldsymbol{F} = \begin{bmatrix} \rho v \\ \rho v u \\ \rho v^2 + P \\ v(\rho e + P) \end{bmatrix} \qquad (2)$$

These equations are derived from the conservation laws: conservation of mass, conservation of momentum, and conservation of energy.

Coordinate Transformation

Equation (1) applies to the physical domain and can be transformed into the computational domain, where it is more convenient to numerically solve. Figure 3 shows both the physical and computational coordinate systems for the nozzle. To accomplish this transformation, the following mapping is applied:

$$\zeta = x \qquad \eta = \eta(x,y) \tag{3}$$

Here, the axial coordinate, *x*, maps directly into the computational domain, but the normal coordinate, y, is transformed by a nonlinear function. To determine what these functions are, the chain rule of multivariable calculus must be applied:

$$\frac{\partial()}{\partial x} = \frac{\partial\zeta}{\partial x}\frac{\partial()}{\partial\zeta} + \frac{\partial\eta}{\partial x}\frac{\partial()}{\partial\eta} = \zeta_x\frac{\partial()}{\partial\zeta} + \eta_x\frac{\partial()}{\partial\eta} \tag{4}$$

$$\frac{\partial()}{\partial y} = \frac{\partial\zeta}{\partial y}\frac{\partial()}{\partial\zeta} + \frac{\partial\eta}{\partial y}\frac{\partial()}{\partial\eta} = \zeta_y\frac{\partial()}{\partial\zeta} + \eta_y\frac{\partial()}{\partial\eta} \tag{5}$$

From Eq (3), it can be seen that:

$$\zeta_x = 1 \qquad\qquad \zeta_y = 0 \tag{6}$$

Combining Eqs (4), (5), and (6) gives the chain rule form that is used to transform the governing equations from the physical to the computational domain:

$$\frac{\partial()}{\partial x} = \frac{\partial()}{\partial \zeta} + \eta_x \frac{\partial()}{\partial \eta} \tag{7}$$

$$\frac{\partial()}{\partial y} = \eta_y \frac{\partial()}{\partial \eta} \tag{8}$$

Applying Eqs (7) and (8) to the original governing equations, Eqs (1) and (2), gives the transformed governing equations in the computational domain:

$$\frac{\partial(\boldsymbol{E})}{\partial \zeta} + \eta_x \frac{\partial(\boldsymbol{E})}{\partial \eta} + \eta_y \frac{\partial(\boldsymbol{F})}{\partial \eta} = 0 \tag{9}$$

Equation (9) can be written in a more convenient form as

$$\frac{\partial(\boldsymbol{E})}{\partial \zeta} = -\eta_x \frac{\partial(\boldsymbol{E})}{\partial \eta} - \eta_y \frac{\partial(\boldsymbol{F})}{\partial \eta} \tag{10}$$

For further information on the governing equations or the transformation of the equations, see Appendix G in reference 3.

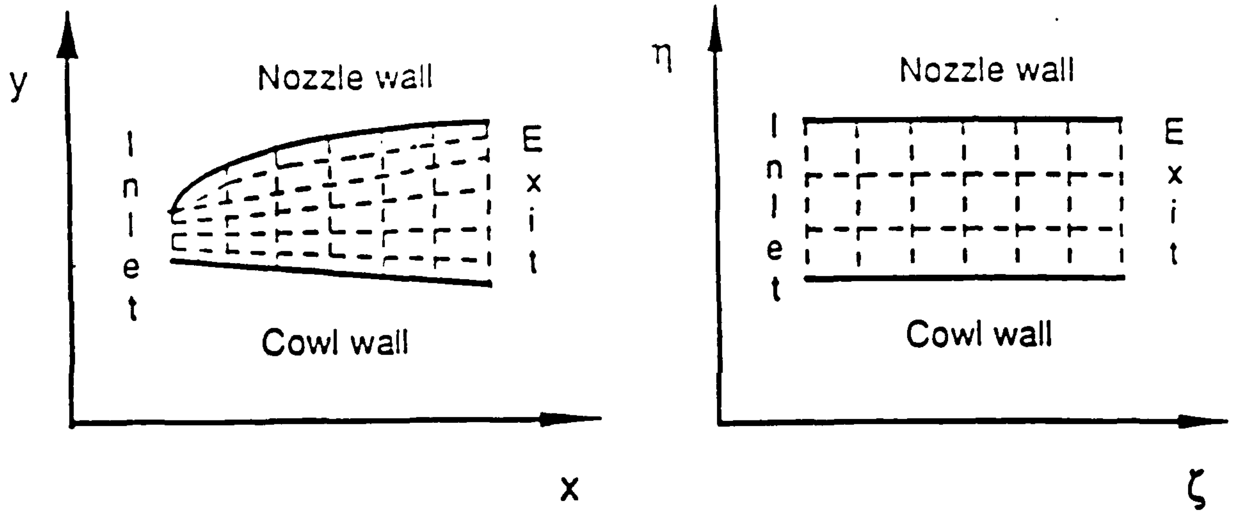


Figure 3. Coordinate Transformation (3:150).

III. FLUX-DIFFERENCE-SPLITTING

Riemann Problem

The Riemann problem is the heart of the FDS method. The flowfield must be discretized before the Riemann problem is applied. The first step in the process is to model the general flow property, ψ in Figure 4a, which has an arbitrary spatial distribution, as a series of nodes. Each node represents a localized region of uniform flow. The Riemann problem assumes that these localized uniform flow regions, at j and j+1 in Figure 4a, extend to a distance half-way between the two nodes, and a discontinuity is assumed to exist at the midpoint, j+1/2.

Waves are generated by the discontinuity at j+1/2. These waves can be seen in Figure 4b. Waves (1) and (3) can either be shock waves, expansion waves, or a combination of one of each type of wave depending on the local flow configuration. Wave (2) is a contact surface. These three waves separate the flow into four regions of uniform flow, with regions 6 and 0 having values equal to known values at nodes above and below the Riemann location, respectively. Regions 2 and 4 correspond to unknown values at the Riemann location that must be solved for.

Three different methods exist to solve the Riemann problem (3:10-13). The first method is an exact solution to the Riemann problem where an iteration is performed using the oblique shock wave relationships across shock waves, and Prandtl-Meyer expansion relationships across expansion waves. The second method approximates the shock wave as an isentropic compression and again requires iteration using Prandtl-Meyer relationships across both expansions and compressions. The third method is a linearized approximate solution that treats the local Riemann problem as isentropic. This method uses a linearized form of the Prandtl-Meyer relationships to analytically solve across both expansions and compressions. Only the exact solution is used in this research.

Exact Solution to the Riemann Problem. The Riemann problem is solved exactly for the properties in regions 2 and 4, as shown in Figure 4b, by using oblique shock and Prandtl-Meyer expansion relationships. This is an iterative process in which the parameters in regions 2 and 4 are calculated separately, but there are two known conditions. First, the pressure in region 4 must match that of region 2 since the contact surface cannot support a normal pressure gradient. The second known condition is that the flow angle in region 4 is equal to the flow angle in region 2 with the two angles equal to that of the contact surface.

Shock Wave Relationships. For the example in Figure 4, the pressure at j+1 is greater than the pressure at node j. Therefore, wave 1 is a shock wave, and must be solved using the oblique shock wave relationships. The following shock wave relationships are written with 0 corresponding to known values upstream in Riemann region 0, and 2 corresponding to values that must be solved for downstream in Riemann region 2. In the oblique shock example shown in Figures 4 and 5, the pressure in region 2 is greater than the pressure in region 0. The following oblique shock wave equations are taken from Zucrow (9:359-360).

$$\delta_2 = \theta_2 - \theta_0 \qquad (11)$$

$$\frac{1}{\tan(\delta_2)} = \left(\frac{\gamma + 1}{2} \frac{M_0^2}{M_0^2 \sin^2(\varepsilon) - 1} - 1 \right) \tan(\varepsilon) \qquad (12)$$

$$\beta = \varepsilon - \delta_2 \qquad (13)$$

$$\frac{P_2}{P_0} = \frac{2\gamma}{\gamma + 1} M_0^2 \sin^2(\varepsilon) - \frac{\gamma - 1}{\gamma + 1} \qquad (14)$$

$$\frac{\rho_2}{\rho_0} = \frac{\tan(\varepsilon)}{\tan(\beta)} = \frac{(\gamma + 1) M_0^2 \sin^2(\varepsilon)}{2 + (\gamma - 1) M_0^2 \sin^2(\varepsilon)} \qquad (15)$$

$$\frac{V_2}{V_0} = \frac{\sin(\varepsilon)}{\sin(\beta)} \left[\frac{2}{(\gamma + 1) M_0^2 \sin^2(\varepsilon)} + \frac{\gamma - 1}{\gamma + 1} \right] \tag{16}$$

$$\frac{\tan(\varepsilon)}{\tan(\beta)} = \frac{2}{\gamma + 1} \left(\frac{1}{M_2^2 \sin^2(\beta)} + \frac{\gamma - 1}{2} \right) \tag{17}$$

$$u_2 = V_2 \cos(\theta_2) \tag{18}$$

$$v_2 = V_2 \sin(\theta_2) \tag{19}$$

Expansion Wave Relationships. For the example, we are assuming isentropic expansion, and using Prandtl-Meyer expansion relationships. In this analysis, the subscripts correspond to the applicable Riemann region. See Figures 6 and 7 for a graphical representation of the example. The following equations are used across an expansion fan:

$$\delta_4 = \theta_4 - \theta_6 \tag{20}$$

$$b = \left[\frac{\gamma + 1}{\gamma - 1} \right]^{1/2} \tag{21}$$

$$\nu_6 = b \arctan\left[\frac{1}{b}\sqrt{M_6^2 - 1}\right] - \arctan\left[\sqrt{M_6^2 - 1}\right] \qquad \textbf{(22)}$$

$$\nu_4 = \nu_6 - \delta_4 = b \arctan\left[\frac{1}{b}\sqrt{M_4^2 - 1}\right] - \arctan\left[\sqrt{M_4^2 - 1}\right] \qquad \textbf{(23)}$$

$$\frac{P_4}{P_6} = \left[\frac{1 + \dfrac{\gamma - 1}{2} M_6^2}{1 + \dfrac{\gamma - 1}{2} M_4^2}\right]^{\frac{\gamma}{(\gamma - 1)}} \qquad \textbf{(24)}$$

$$\frac{\rho_4}{\rho_6} = \left[\frac{P_4}{P_6}\right]^{\frac{1}{(\gamma - 1)}} \qquad \textbf{(25)}$$

$$a_4 = \left[\frac{\gamma \; P_4}{\rho_4}\right]^{1/2} \qquad \textbf{(26)}$$

$$u_4 = M_4 \; a_4 \; \cos(\theta_4) \qquad \textbf{(27)}$$

$$v_4 = M_4 \; a_4 \; \sin(\theta_4) \qquad \textbf{(28)}$$

Remember, as stated above, in Riemann regions 2 and 4 the flow angles and pressures must match:

$$\theta_4 = \theta_2 \tag{29}$$

$$P_4 = P_2 \tag{30}$$

This is an iterative process, and the flow turning angles may change from compression to expansion, expansion to compression, or experience no turning. Therefore, the flow turning angle should be checked after each iteration to determine the analytical mode that will be used next. The iteration process is as follows (3:166-167):

1. The initial conditions for regions 6 and 0 are known and remain fixed.
2. The flow angle for wave (2) is guessed. (The average of the flow angles between regions 6 and 0 is usually sufficient as a first guess).
3. Solve the shock wave problem for the pressure in region 2.

   a. The flow deflection angle is calculated as the change in the flow angle between regions 0 and 2 for the shock from Eq (11).

   b. For the assumed turning angle, the oblique shock wave angle is iteratively calculated from Eq (12).

   c. The static pressure in region 2 is then calculated from Eq (14).

4. Solve the expansion wave problem from regions 6 to 4.

a. Using Eq (29) the flow angle in region 4 is required to be the same as the flow angle in region 2.

b. The flow deflection angle is then calculated as the change in the flow angle between regions 6 and 4 for the expansion wave using Eq (20).

c. The Prandtl-Meyer angle is calculated for region 6 using Eq (22) and remains fixed.

d. The Prandtl-Meyer angle is calculated for region 4 using the first of Eqs (23).

e. The Mach number in region 4 is iteratively calculated from the second of Eqs (23).

f. The static pressure in region 4 is calculated from Eq (24).

5. Check for consistent solutions.

a. The static pressure in regions 4 and 2 must match across the contact surface according to Eq (30).

b. If the pressures are equal (or within a specified tolerance), the iteration is complete.

c. If the pressures are not equal, a new flow angle is guessed and the iteration continues from step 3.

6. Calculate remaining properties from the shock and expansion wave relations.

Flux Differencing

Before the fluxes can be differenced, they must be computed. This is accomplished by using the primitive variables that were calculated by the exact solution to the Riemann problem. Recall that these primitive variables were computed at the Riemann locations, which are positioned halfway between the nodes, as shown in Figure 8. The Riemann fluxes must be calculated for all four components of the ***E*** and ***F*** vectors, in each of the four Riemann regions, and are recombined as defined by Eqs (1) and (2), repeated here for convenience:

$$\frac{\partial \boldsymbol{E}}{\partial x} + \frac{\partial \boldsymbol{F}}{\partial y} = 0 \qquad \textbf{(1)}$$

where

$$\boldsymbol{E} = \begin{bmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ u(\rho e + P) \end{bmatrix} \qquad \boldsymbol{F} = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + P \\ v(\rho e + P) \end{bmatrix} \qquad \textbf{(2)}$$

As an example, the first *E* flux component, *E*1, would be computed at each Riemann location for all four Riemann regions:

$$(E1)_0 = \rho_0 u_0 \tag{31}$$

$$(E1)_2 = \rho_2 u_2 \tag{32}$$

$$(E1)_4 = \rho_4 u_4 \tag{33}$$

$$(E1)_6 = \rho_6 u_6 \tag{34}$$

After the fluxes have been calculated at all the Riemann locations on the vertical plane, see Figure 8, they must be locally differenced *across* waves (1), (2), and (3). Differencing *E*1 across wave (1) corresponds to differencing the density/axial velocity product between regions 0 and 2. This can be shown mathematically as:

$$(dE1)_{wave\ 1} = (E1)_2 - (E1)_0 = \rho_2 u_2 - \rho_0 u_0 \tag{35}$$

Likewise, differencing *E*1 across waves (2) and (3):

$$(dE1)_{wave\ 2} = (E1)_4 - (E1)_2 = \rho_4 u_4 - \rho_2 u_2 \tag{36}$$

$$(dE1)_{wave\ 3} = (E1)_6 - (E1)_4 = \rho_6 u_6 - \rho_4 u_4 \qquad \textbf{(37)}$$

Summing the flux differences across all three waves gives the total flux difference at the Riemann location, j+1/2:

$$(dE1)_{j+1/2} = [\ (dE1)_{wave\ 3} + (dE1)_{wave\ 2} + (dE1)_{wave\ 1}\ ]_{j+1/2} \qquad \textbf{(38)}$$

The differences of the other three *E* components and the four *F* components are computed in a similar fashion.

At this point, the FDS approach is still similar to the finite-difference approach in that Eq (38) represents the total difference between nodes j and j+1. This is where the FDS method differs from the finite-difference method in that for FDS the differences are split into positively and negatively biased components.

<u>Splitting the Flux Differences</u>

Splitting the flux differences is a directional biasing of the flux differences. As a result, only relevant information is received at each downstream node, as illustrated in Figure 9.

Splitting the flux differences is accomplished by breaking up the flux differences, that were calculated at each Riemann location, into positive and negative components. For the *E*1 flux difference, splitting across each wave would give:

$$(dE1)_{wave\ 1} = (dE1^{+})_{wave\ 1} + (dE1^{-})_{wave\ 1} \tag{39}$$

$$(dE1)_{wave\ 2} = (dE1^{+})_{wave\ 2} + (dE1^{-})_{wave\ 2} \tag{40}$$

$$(dE1)_{wave\ 3} = (dE1^{+})_{wave\ 3} + (dE1^{-})_{wave\ 3} \tag{41}$$

Here, the "+" superscript indicates that this quantity is the portion of the flux difference that passes information in the positive y direction. Inversely, the "-" superscript only passes information in the negative y direction. As before, the split flux differences can be added over all three waves to get the total positive and negative flux differences at the Riemann location for all *E* and *F* components.

$$(dE1)^{+}_{j+1/2} = \left((dE1)^{+}_{wave\ 1} + (dE1)^{+}_{wave\ 2} + (dE1)^{+}_{wave\ 3}\right)_{j+1/2} \tag{42}$$

$$(dE1)^{-}_{j+1/2} = \left((dE1)^{-}_{wave\ 1} + (dE1)^{-}_{wave\ 2} + (dE1)^{-}_{wave\ 3}\right)_{j+1/2} \tag{43}$$

Again, this procedure would be repeated for the other three *E* and four *F* vector components.

Values are known at computational plane i, and the split flux differences are used to pass information to the next plane, i+1. See Figure 9 for a graphical representation. The information is passed in the direction of characteristics and streamlines in the computational domain. These characteristic slopes are given by (dy/dx) in the physical domain, but in the computational domain they are:

$$\begin{aligned}\Lambda_1 = \frac{d\eta}{d\zeta} = \frac{d\eta}{dx} &= \frac{\partial\eta}{\partial x}\frac{dx}{dx} + \frac{\partial\eta}{\partial y}\frac{dy}{dx} \\ &= \eta_x + \eta_y\left[\frac{dy}{dx}\right]_1\end{aligned} \tag{44}$$

$$\Lambda_2 = \eta_x + \eta_y\left[\frac{dy}{dx}\right]_2 \tag{45}$$

$$\Lambda_3 = \eta_x + \eta_y\left[\frac{dy}{dx}\right]_3 \tag{46}$$

where

$$\left[\frac{dy}{dx}\right]_1 = \frac{uv - a^2\sqrt{M^2-1}}{u^2 - a^2} \tag{47}$$

$$\left[\frac{dy}{dx}\right]_2 = \frac{v}{u} \qquad (48)$$

$$\left[\frac{dy}{dx}\right]_3 = \frac{uv + a^2\sqrt{M^2 - 1}}{u^2 - a^2} \qquad (49)$$

Equations (47), (48), and (49) can be manipulated to give the characteristic slopes in the physical domain in terms of flow angle and Mach angle, see Doty (4):

$$\left[\frac{dy}{dx}\right]_1 = \tan(\theta - \alpha) \qquad (50)$$

$$\left[\frac{dy}{dx}\right]_2 = \tan(\theta) \qquad (51)$$

$$\left[\frac{dy}{dx}\right]_3 = \tan(\theta + \alpha) \qquad (52)$$

Maximum Step Size for Marching Algorithm

The step size, for this steady problem, corresponds to the largest axial distance, from plane i to plane i+1, that can be made while keeping the solution stable, and must be determined at each computational plane. According to Anderson et al. (1:76), the CFL condition requires that the analytic domain of influence must lie within the numerical domain of influence, as illustrated in Figure 10. Therefore, any step size larger than this amount would cause instability. After all the characteristic slopes and stream line locations have been computed at the computational plane, i, a local step size, $\Delta\zeta$, can be determined for each nodal location, j. The local step size is equal to the minimum axial distance of the intersection of the streamline from node j and either the negatively biased characteristic at j+1 or the positively biased characteristic from j-1. The magnitude of $\Delta\zeta$ for the computational plane, i, is equal to the minimum local $\Delta\zeta$ on that plane. Also, for this research, a Courant number multiplier of 0.99 was used.

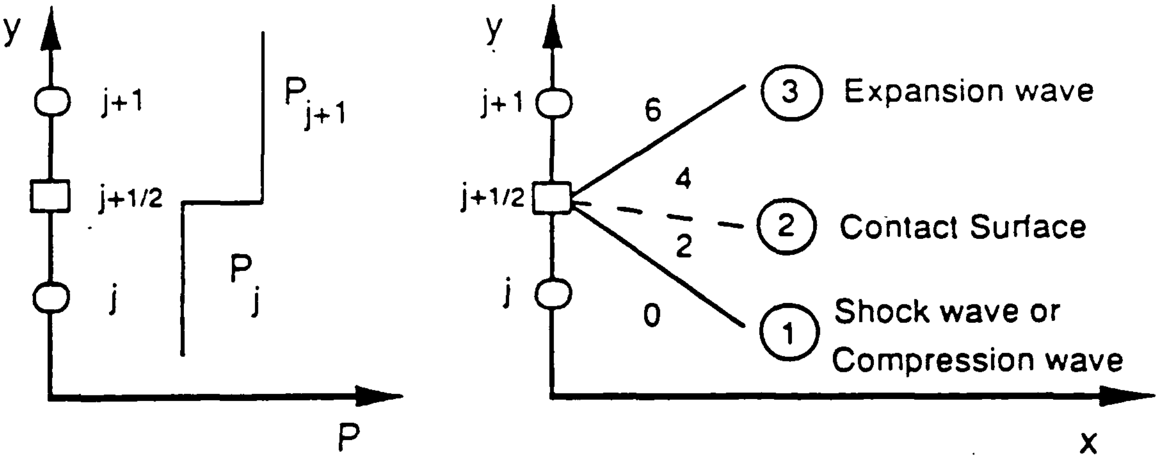


Figure 4. Riemann problem for planar supersonic flow (3:14).

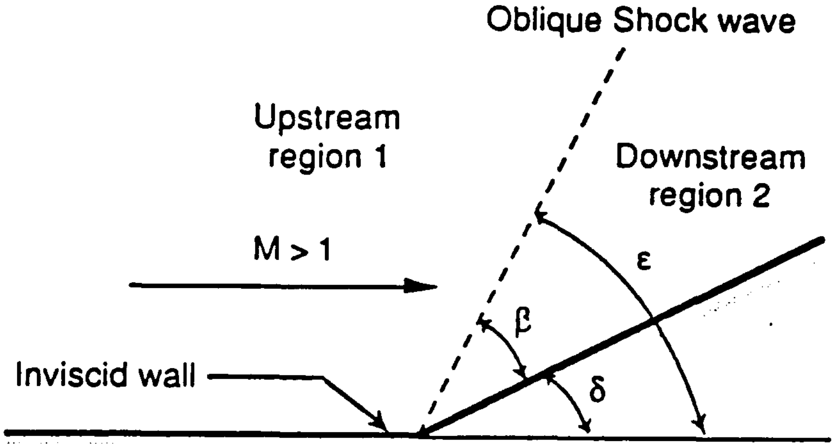


Figure 5. Oblique shock wave geometry (3:178).

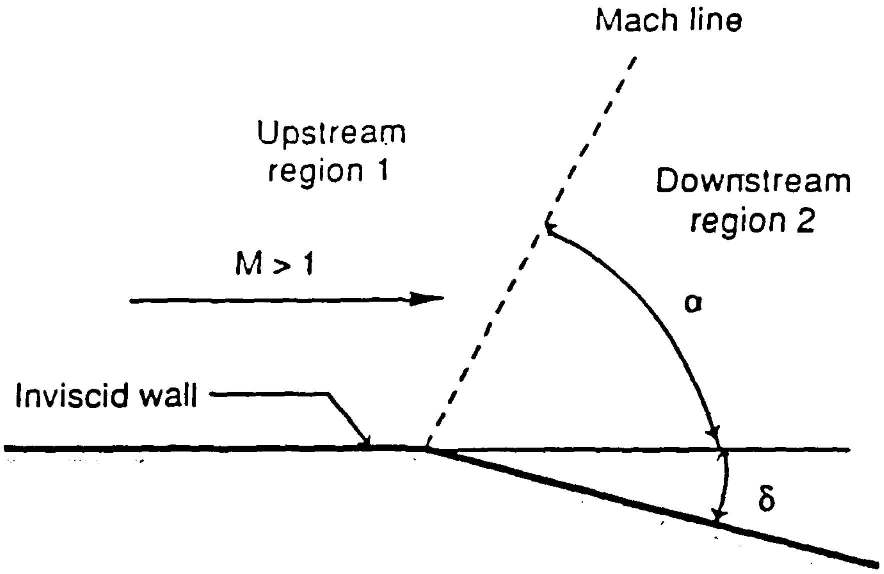
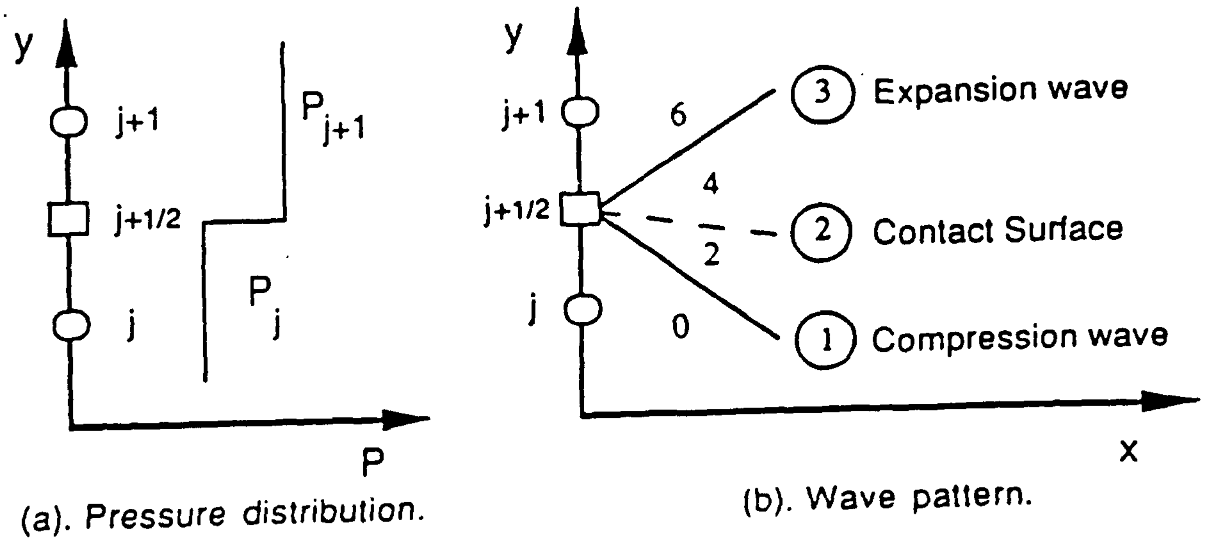


Figure 6. Prandtl-Meyer expansion wave geometry (3:179).



(a). Pressure distribution.

(b). Wave pattern.

Figure 7. Specific Riemann problem (3:179).

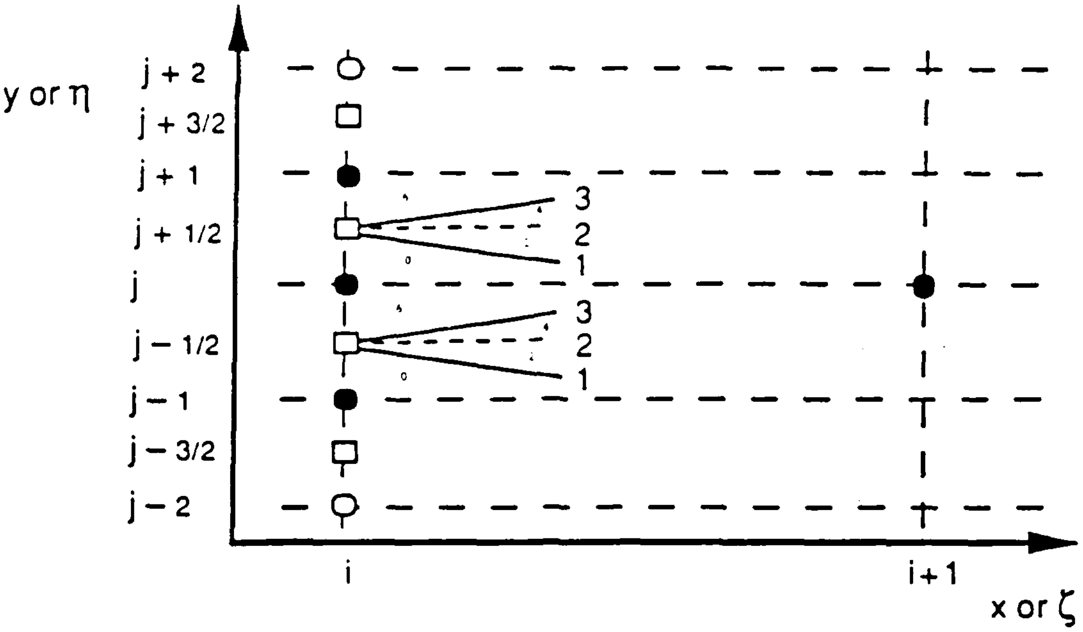


Figure 8. Flux differences and splitting (3:188).

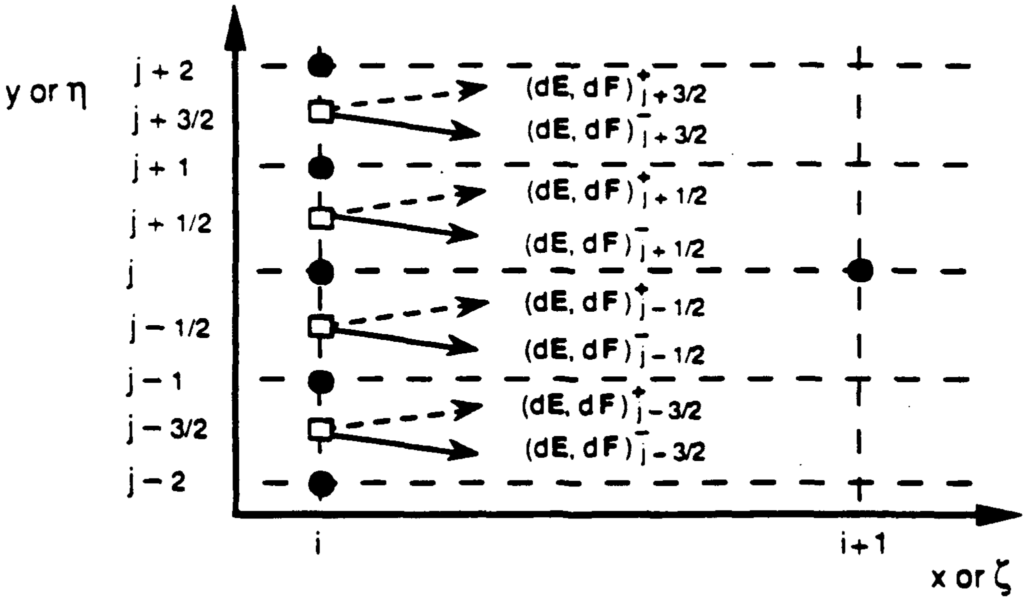


Figure 9. Directional information passed from plane i to plane i+1 (3:212).

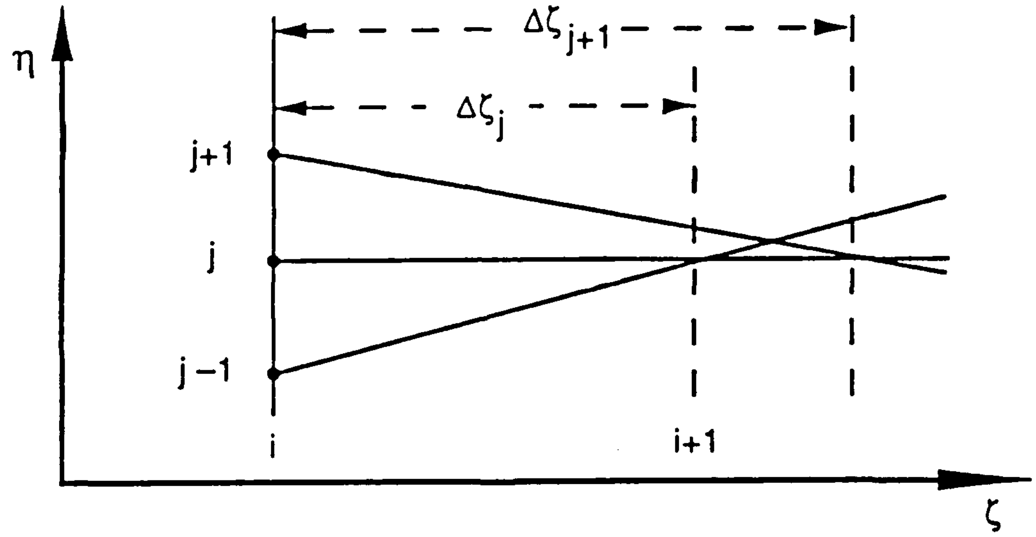


Figure 10. Step size determination (3:157).

IV. First-Order Accurate Upwind Flux-Difference-Split Numerical Algorithm

First-Order Accurate Interior Point FDS Approximations

Recall that the governing equations for steady, inviscid, adiabatic, planar flow in computational space are given by:

$$\frac{\partial(\boldsymbol{E})}{\partial\zeta} = -\eta_x\frac{\partial(\boldsymbol{E})}{\partial\eta} - \eta_y\frac{\partial(\boldsymbol{F})}{\partial\eta} \tag{10}$$

These equations are operated on by a flux-difference-splitting in the normal direction, and finite differencing in the axial direction. The governing equations can be rewritten into their approximate computational form, shown in Eq (53), (3:190).

$$\frac{\Delta_i(\boldsymbol{E})}{\Delta\zeta} = -\eta_x\frac{\Delta_j(\boldsymbol{E})}{\Delta\eta} - \eta_y\frac{\Delta_j(\boldsymbol{F})}{\Delta\eta} \tag{53}$$

Here, for computational convenience,

$$\Delta\eta = 1 \tag{54}$$

Substituting Eq (54) into Eq (53), and rearranging gives:

$$\Delta_i(\boldsymbol{E}) = -\Delta\zeta\ \eta_x\ \Delta_j(\boldsymbol{E})\ -\Delta\zeta\ \eta_y\ \Delta_j\ (\boldsymbol{F}) \qquad (55)$$

The "*i*" subscript indicates a finite-difference operator in the axial direction. In this case, a two-point, upwind, first-order accurate, finite-difference operator is applied between planes i and i+1. The "*j*" subscript, in Eq (55), signifies that a FDS operator is applied in the normal direction. Here, a two-point, upwind, first-order accurate, FDS operator is applied in both the positive and negative y directions at node j on plane i, resulting in a three point stencil in the normal direction, as shown in Figure 11. In Eq (55), the finite-difference operator is:

$$\Delta_i(\boldsymbol{E}) = \boldsymbol{E}_j^{i+1} - \boldsymbol{E}_j^{i} \qquad (56)$$

The flux-difference operators are:

$$\Delta_j(\boldsymbol{E}) = \left[d\boldsymbol{E}^{+}_{j-1/2} + d\boldsymbol{E}^{-}_{j+1/2}\right] \qquad (57)$$

and

$$\Delta_j(\boldsymbol{F}) = \left[d\boldsymbol{F}^{+}_{j-1/2} + d\boldsymbol{F}^{-}_{j+1/2}\right] \qquad (58)$$

Note how only the positive biased information from the Riemann location below j, at j-1/2, and the negative biased information from the Riemann location above, at j+1/2, is used to determine the new values at the next computational plane, i+1. Substituting Eqs (56), (57), and (58) into Eq (55) and rearranging gives:

$$\begin{aligned} \boldsymbol{E}_j^{i+1} = \boldsymbol{E}_j^{i} - \Delta\zeta\ \eta_x \left[d\boldsymbol{E}_{j-1/2}^{+} + d\boldsymbol{E}_{j+1/2}^{-}\right] \\ - \Delta\zeta\ \eta_y \left[d\boldsymbol{F}_{j-1/2}^{+} + d\boldsymbol{F}_{j+1/2}^{-}\right] \end{aligned} \qquad (59)$$

Equation (59) represents the first-order differencing equation in chain rule conservation form. The chain rule conservation form uses metrics that are calculated at the nodes. The weak conservation law form, which is another method for solving the governing equations, can be found in Appendix A.

First-Order Accurate Boundary Point FDS Approximations

When calculating values at the upper boundary, of plane i+1, there is no Riemann location above the boundary to use, as can be seen in Figure 12. Therefore, only physical information at Riemann location j-1/2 can be used. Thus, the split flux differences coming from j+1/2 do not exist and are deleted from Eq (59) for the upper boundary. For the upper boundary, Eq (59) becomes:

$$\boldsymbol{E}_j^{i+1} = \boldsymbol{E}_j^{i} - \Delta\zeta\ \eta_x\,(d\boldsymbol{E}^{+})_{j-1/2} - \Delta\zeta\ \eta_y\,(d\boldsymbol{F}^{+})_{j-1/2} \tag{60}$$

For the lower boundary:

$$\boldsymbol{E}_j^{i+1} = \boldsymbol{E}_j^{i} - \Delta\zeta\ \eta_x\,(d\boldsymbol{E}^{-})_{j+1/2} - \Delta\zeta\ \eta_y\,(d\boldsymbol{F}^{-})_{j+1/2} \tag{61}$$

After the partial solution at plane i+1 has been computed using Eqs (60) and (61), the conservative variables are decoded into the primitive variables ρ, u, v, p, and ρe. The resultant decoded solution does not necessarily obey the inviscid velocity tangency condition at the wall. Therefore, to be physically consistent with the geometry, the primitive variables need to be corrected, and a wave corrector is applied to the solution to turn the flow parallel to the wall. To accomplish this turning, a new deflection angle, δ, is specified as:

$$\begin{gathered} \delta = \theta_{sol} - \theta_{wall} \\ \textit{where} \\ \theta_{sol}\ \textit{is the computed flow angle} \\ \theta_{wall}\ \textit{is the wall angle} \end{gathered} \tag{62}$$

After the deflection angle is known, the flow can be turned parallel to the wall by one of two methods. If a compression wave is required to turn the flow, the oblique shock wave relationships are used. A shock wave angle, ε, can be iterated using Eq (12), and the corrected pressure, density, and velocity components are computed. If and expansion is required to turn the flow, the Prandtl-Meyer expansion wave relationships are used. Equation (23) would be iterated to determine the downstream Mach number. Once the Mach number is known, the primitive variables can be computed.

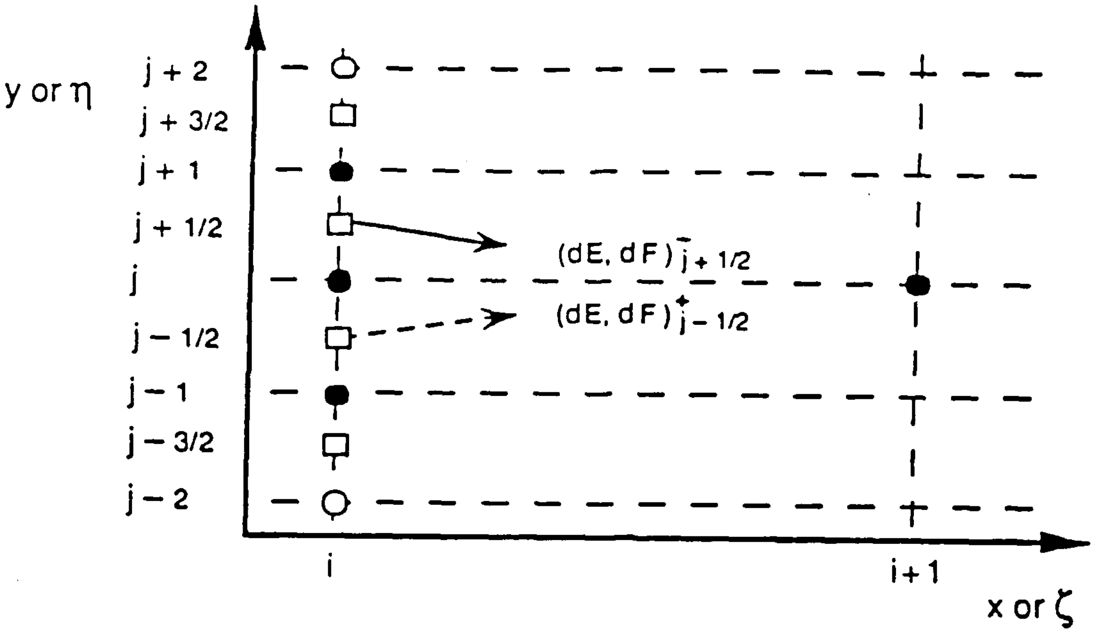


Figure 11. Stencil for first-order accurate FDS upwind method (3:196).

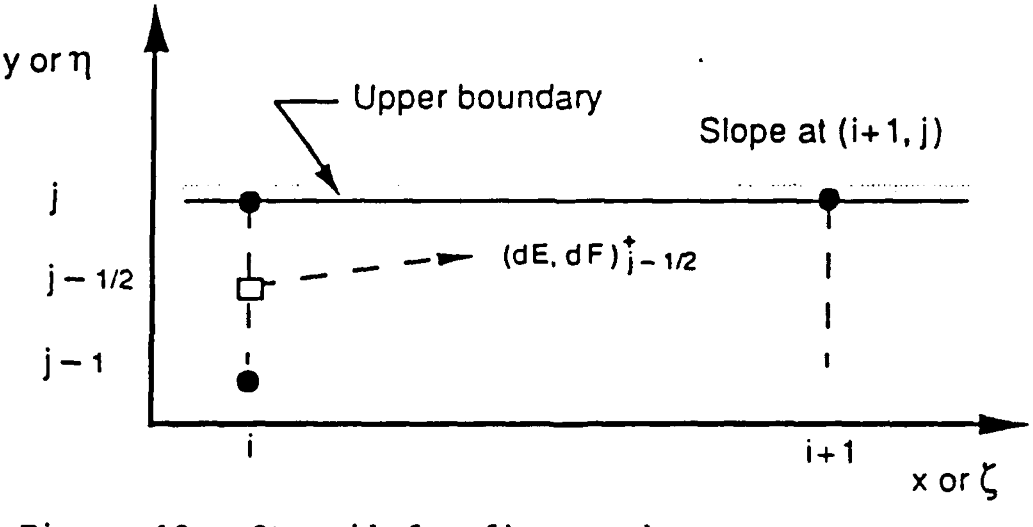


Figure 12. Stencil for first-order accurate upper solid wall boundary point (3:196).

V. Second-Order Accurate Center-Spaced Flux-Difference-Split Numerical Algorithm

Introduction

The first-order accurate method is the basis upon which the second-order accurate center-spaced method is built. Recall that the solution is advanced with first-order accuracy using Eq (59):

$$\begin{aligned} \boldsymbol{E}_j^{i+1} = \boldsymbol{E}_j^i &- \Delta\zeta\ \eta_x \left[d\boldsymbol{E}_{j-1/2}^{+} + d\boldsymbol{E}_{j+1/2}^{-}\right] \\ &- \Delta\zeta\ \eta_y \left[d\boldsymbol{F}_{j-1/2}^{+} + d\boldsymbol{F}_{j+1/2}^{-}\right] \end{aligned} \qquad (59)$$

Equation (59) can be rewritten as

$$\boldsymbol{E}_j^{i+1} = \boldsymbol{E}_j^i + \Delta_j(\boldsymbol{E})_{1xy} + \Delta_j(\boldsymbol{F})_{1xy} \qquad (63)$$

where

$$\Delta_j(\boldsymbol{E})_{1xy} \equiv -\ \Delta\zeta\ \eta_x \left[(d\boldsymbol{E}^{+})_{j-1/2} + (d\boldsymbol{E}^{-})_{j+1/2}\right] \qquad (64)$$

$$\Delta_j(\boldsymbol{F})_{1xy} \equiv -\ \Delta\zeta\ \eta_y \left[(d\boldsymbol{F}^{+})_{j-1/2} + (d\boldsymbol{F}^{-})_{j+1/2}\right] \qquad (65)$$

In the above equations, the *1xy* subscript indicates that these quantities are first-order accurate corrections in the x and *y* directions, and are functions of the sums of the flux differences.

Second-Order Accurate Center-Spaced Interior Point FDS Approximations

Second-order accuracy can be implemented by two different methods. First, and most obvious, is to use a second-order accurate differencing method for the derivative approximations in the governing equations. The second method, which is used here, uses a first-order accurate solution and adds a second-order corrector to it. Thus, the second-order accurate solution is comprised of the first-order accurate solution and second-order corrections.

Doty has shown that a second-order correction applied to the linear hyperbolic convection equation, which is often used to model the essential characteristics of the nonlinear system of equations, has a modified equation that demonstrates second-order accuracy (3:108). For a complete description of this process, refer to Appendices A, B, and L in reference 3.

The second-order accurate equation can be written in shorthand notation as:

$$\begin{aligned} \boldsymbol{E}_j^{i+1} = \boldsymbol{E}_j^{i} &+ \Delta_j(\boldsymbol{E})_{1xy} + \Delta_j(\boldsymbol{E})_{2y} + \Delta_j(\boldsymbol{E})_{2x} \\ &+ \Delta_j(\boldsymbol{F})_{1xy} + \Delta_j(\boldsymbol{F})_{2y} + \Delta_j(\boldsymbol{F})_{2x} \end{aligned} \tag{66}$$

The second-order normal corrections are:

$$
\begin{aligned}
\Delta_j(\boldsymbol{E})_{2y} \equiv & - \frac{1}{2}\,\Delta\zeta\;\eta_x \left[(d\boldsymbol{E}^{+})_{j+1/2} - (d\boldsymbol{E}^{+})_{j-1/2}\right] \\
& + \frac{1}{2}\,\Delta\zeta\;\eta_x \left[(d\boldsymbol{E}^{-})_{j+1/2} - (d\boldsymbol{E}^{-})_{j-1/2}\right]
\end{aligned}
\tag{67}
$$

and

$$
\begin{aligned}
\Delta_j(\boldsymbol{F})_{2y} \equiv & - \frac{1}{2}\,\Delta\zeta\;\eta_y \left[(d\boldsymbol{F}^{+})_{j+1/2} - (d\boldsymbol{F}^{+})_{j-1/2}\right] \\
& + \frac{1}{2}\,\Delta\zeta\;\eta_y \left[(d\boldsymbol{F}^{-})_{j+1/2} - (d\boldsymbol{F}^{-})_{j-1/2}\right]
\end{aligned}
\tag{68}
$$

The second-order axial corrections are:

$$
\begin{aligned}
\Delta_j(\boldsymbol{E})_{2x} = & \frac{1}{2}\,\Delta\zeta^2\;\eta_x \Big[\big((\Lambda\;d\boldsymbol{E}^{+})_{j+1/2} - (\Lambda\;d\boldsymbol{E}^{+})_{j-1/2}\big)_1 \\
& \qquad + \big((\Lambda\;d\boldsymbol{E}^{+})_{j+1/2} - (\Lambda\;d\boldsymbol{E}^{+})_{j-1/2}\big)_2 \\
& \qquad + \big((\Lambda\;d\boldsymbol{E}^{+})_{j+1/2} - (\Lambda\;d\boldsymbol{E}^{+})_{j-1/2}\big)_3\Big] \\
& - \frac{1}{2}\,\Delta\zeta^2\;\eta_x \Big[\big((|\Lambda|\;d\boldsymbol{E}^{-})_{j+1/2} - (|\Lambda|\;d\boldsymbol{E}^{-})_{j-1/2}\big)_1 \\
& \qquad + \big((|\Lambda|\;d\boldsymbol{E}^{-})_{j+1/2} - (|\Lambda|\;d\boldsymbol{E}^{-})_{j-1/2}\big)_2 \\
& \qquad + \big((|\Lambda|\;d\boldsymbol{E}^{-})_{j+1/2} - (|\Lambda|\;d\boldsymbol{E}^{-})_{j-1/2}\big)_3\Big]
\end{aligned}
\tag{69}
$$

and

$$
\begin{aligned}
\Delta_j (\boldsymbol{F})_{2x} = \frac{1}{2}\, \Delta\zeta^2\, \eta_y \Big[& \big((\Lambda\ d\boldsymbol{F}^+)_{j+1/2} - (\Lambda\ d\boldsymbol{F}^+)_{j-1/2}\big)_1 \\
& + \big((\Lambda\ d\boldsymbol{F}^+)_{j+1/2} - (\Lambda\ d\boldsymbol{F}^+)_{j-1/2}\big)_2 \\
& + \big((\Lambda\ d\boldsymbol{F}^+)_{j+1/2} - (\Lambda\ d\boldsymbol{F}^+)_{j-1/2}\big)_3 \Big] \\
- \frac{1}{2}\, \Delta\zeta^2\, \eta_y \Big[& \big((|\Lambda|\ d\boldsymbol{F}^-)_{j+1/2} - (|\Lambda|\ d\boldsymbol{F}^-)_{j-1/2}\big)_1 \\
& + \big((|\Lambda|\ d\boldsymbol{F}^-)_{j+1/2} - (|\Lambda|\ d\boldsymbol{F}^-)_{j-1/2}\big)_2 \\
& + \big((|\Lambda|\ d\boldsymbol{F}^-)_{j+1/2} - (|\Lambda|\ d\boldsymbol{F}^-)_{j-1/2}\big)_3 \Big]
\end{aligned}
\tag{70}
$$

Note that the second-order corrections are differences of the split flux differences, and both the axial and normal correction are required to achieve overall second-order accuracy. As can be seen in Figure 13, the second-order corrections do not pass information in the same manner as the first-order correction. The second-order method passes positive information from node j+1/2, and negative information from node j-1/2. This violates the rules that only positive biased information is used from the Riemann location below node j, and only negative biased information is used from the Riemann location above node j. This is not a problem due to the small magnitude, and influence, of second-order correction terms, Pandolfi (7:606).

Second-Order Accurate Center-Spaced Boundary Point FDS Approximations

The second-order boundary correction terms are computed in a manner similar to the first-order boundary solution. The difference is that information is required at a fictitious Riemann location just outside the boundary, at j+1/2, for the upper boundary in Figure 13. This fictitious location is assigned information so that a difference of a split flux difference can be computed for the Riemann location just inside the boundary, at j-1/2. As an example on the upper boundary, to get a difference of a split flux difference at j-1/2, the split flux differences at locations j-1/2 and j-3/2 must be extrapolated to j+1/2. According to Pandolfi (7:607), a linear extrapolation is sufficient:

$$(d\boldsymbol{E}^{+})_{j+1/2} = 2\ (d\boldsymbol{E}^{+})_{j-1/2} - (d\boldsymbol{E}^{+})_{j-3/2} \tag{71}$$

$$(d\boldsymbol{F}^{+})_{j+1/2} = 2\ (d\boldsymbol{F}^{+})_{j-1/2} - (d\boldsymbol{F}^{+})_{j-3/2} \tag{72}$$

On the lower boundary:

$$(d\boldsymbol{E}^{-})_{j-1/2} = 2\ (d\boldsymbol{E}^{-})_{j+1/2} - (d\boldsymbol{E}^{-})_{j+3/2} \tag{73}$$

$$(d\boldsymbol{F}^{-})_{j-1/2} = 2\ (d\boldsymbol{F}^{-})_{j+1/2} - (d\boldsymbol{F}^{-})_{j+3/2} \tag{74}$$

As an example on the upper boundary, after the split flux differences are determined for the fictitious Riemann location, j+1/2, a difference of a split flux difference can be computed at j-1/2. The second-order correction is computed in a procedure similar to that of the first-order solution, except instead of using split flux differences at j-1/2, the second-order method uses differences of split flux differences at j-1/2 to compute the second-order corrections. Once the second-order correction is known, it is added to the first-order solution. The same method is repeated on the lower boundary. See Chapter IV.

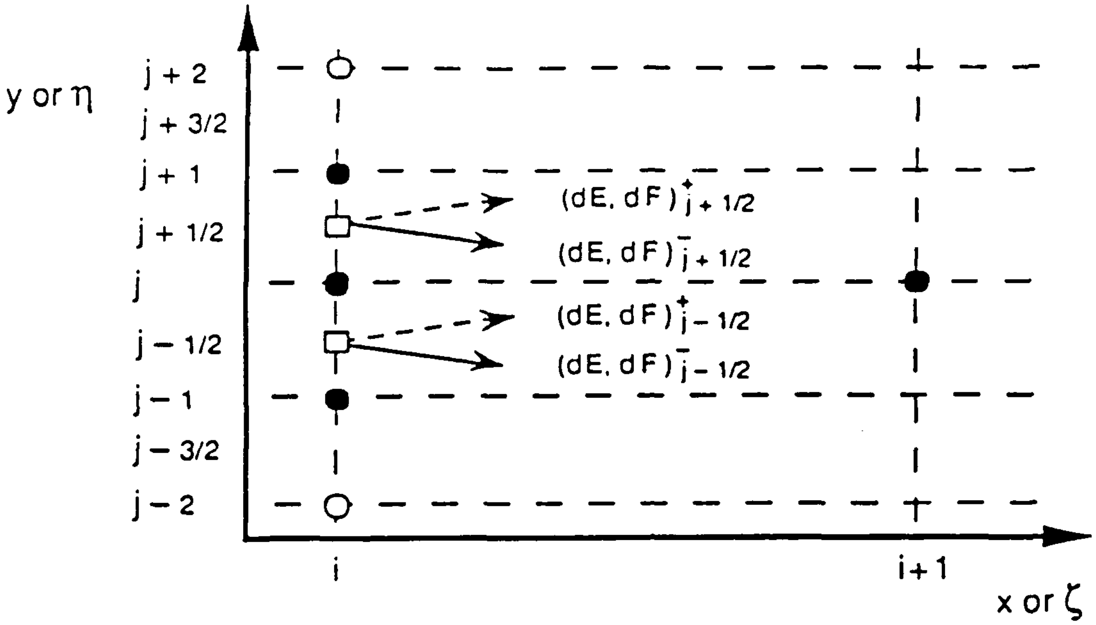


Figure 13. Stencil for second-order accurate center-spaced FDS method (3:213).

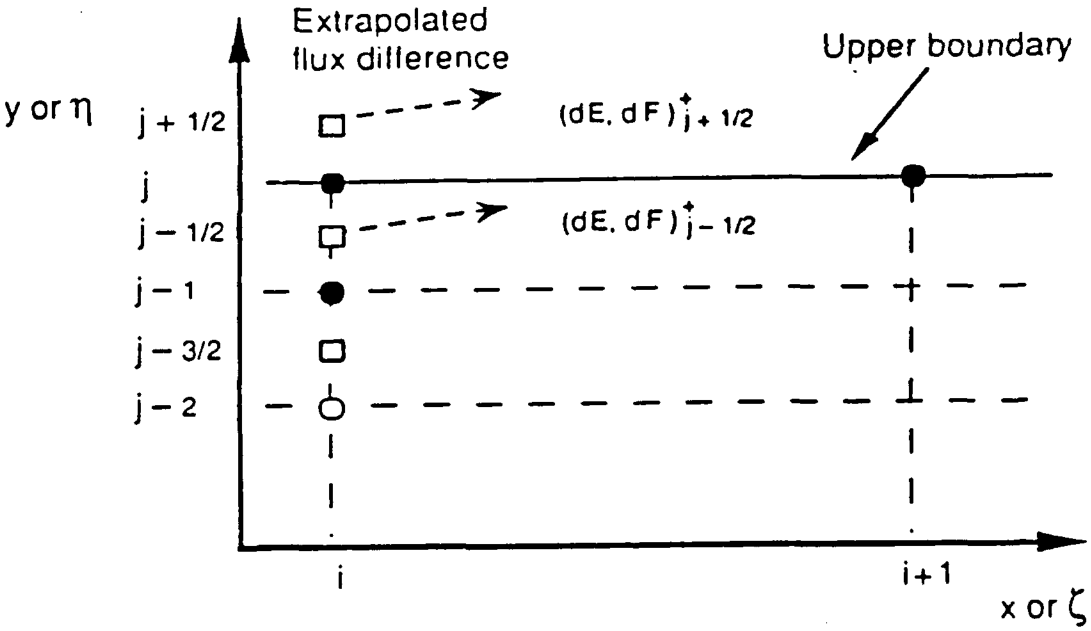


Figure 14. Stencil for second-order accurate center-spaced FDS method at upper boundary (3:214).

VI. Results and Discussion

Two different investigations were performed to determine the effects of the second-order correction terms. The first investigation was a grid refinement study using a supersonic source flow. The second was a method comparison using an oblique shock wave study.

Grid Refinement Study

Before the grid refinement study is discussed, a convention must be defined. In this discussion, methodXX will be used to describe a differencing scheme used in the analysis, where the first X corresponds to the differencing scheme used at an interior location, and the second X would indicate the method used on the boundaries. The definition is as follows:

1- corresponds to a first-order upwind differencing

2- corresponds to a second-order central differencing

Or, stated directly:

method11- 1st order interior, 1st order boundary

method21- 2nd order interior, 1st order boundary

method22- 2nd order interior, 2nd order boundary

These conventions will be used from this point forward.

The grid refinement study was accomplished with a source flow comparison, using the initial conditions in Table I.

Table I. Initial conditions for source flow analysis.

Property	Mach=1.01
static pressure (Pa)	101,325
static temperature (K)	298.0
specific heat ratio	1.4
gas constant (J/Kg/K)	287.0

An example of the source flow geometry, illustrated in Figure 15, consists of an upper and lower wall, each diverging at 15 degrees. The initial plane is at an axial distance of one meter, and the final plane is at four meters. The walls diverged at 15 degrees from the centerline. For an explanation of the exact solution of the source flow, see Appendix O in reference 3.

The normalized static pressure error was computed at each nodal location as follows:

$$\textit{normalized static pressure error} = \frac{P_{numerical}-P_{exact}}{P_{exact}} \quad \textbf{(75)}$$

Interior Point Error Convergence. The interior point error convergence was accomplished by analyzing the flow along a single grid point location that corresponded to the centerline of the source flow for method11, method21, and method22. For each method, a configuration of 11 nodes, 21 nodes, and 31 nodes at each computational plane was studied. The absolute value of the normalized static pressure error was integrated using trapezoidal integration. The results can be seen in Table II.

Table II. Integrated percent error in static pressure along centerline.

	method11	method21	method22
11 nodes	0.847	0.401	0.531
21 nodes	0.394	0.171	0.247
31 nodes	0.234	0.117	0.168

For method11, see Figure 16, increasing the number of nodes from 11 to 21 results in a 53 percent reduction in integrated error. This is consistent with a first-order, finite-difference method.

Figure 17 shows the error convergence for method 21. Again, increasing the number of nodes from 11 to 21 results in a 57 percent reduction in error. While a 57 percent error reduction is slightly better than a first-order, finite-difference convergence, it falls short of the 75 percent reduction that would be achieved for a second-order, finite-differenced equation.

The error convergence for method22 can be seen in Figure 18. Increasing the number of nodes from 11 to 21 results in an integrated error reduction of 53 percent. Interestingly, this is the same reduction seen in method11, and is very close to method22's value. Again, the error reduction falls short of the 75 percent value that would be achieved by a second-order finite differenced equation.

As stated before, the FDS method is similar to finite-differencing up to the point where the flux differences are split. After the flux differences are split, only a portion of the information from a particular Riemann location might be used to compute new values at the next computational plane. A finite-differencing scheme would use all the information, depending on the stencil, in determining the values at the next plane. Because of this splitting, and using different amounts of information from the same location, FDS does not directly relate to finite-differencing.

Boundary Point Error Convergence. The boundary point error convergence was done by looking at the flow along the upper wall of the source flow for method11, method21, and method22. For each method, a configuration of 11 nodes, 21 nodes, and 31 nodes at each computational plane was studied. The results can be seen in Table III.

Table III. Integrated percent error in static pressure along upper boundary.

	method11	method21	method22
11 nodes	3.554	1.658	0.629
21 nodes	1.587	0.771	0.410
31 nodes	1.008	0.482	0.348

For method11, see Figure 19, increasing the number of nodes from 11 to 21 results in a 55 percent reduction in error.

Figure 20 shows the error convergence for method21. Again, increasing the number of nodes from 11 to 21 results in a 53 percent reduction in error. This is approximately the same error convergence as method11 above.

The error convergence for method22 can be seen in Figure 21. Increasing the number of nodes from 11 to 21 results in an integrated error reduction of only 35 percent. Method22 on the boundary is the only method that shows the true error convergence for the second-order boundary since the boundary effects for method22 at the centerline are diminished. This lesser reduction of error at the boundary could be the result of extrapolating a split flux difference to a location outside the boundary, and weighing it evenly with split flux difference information coming from a Riemann location just inside the boundary. In effect, nonphysical information is being used equally with physical information near the boundary.

In summary of both interior and boundary integrated error comparison, all three methods appear to have a near first-order finite-difference convergence behavior. The expected 75 percent error reduction for a second-order method was not demonstrated for the FDS method. This is caused by the splitting of the fluxes, and using the information differently than a finite-differencing method would use it. As a result of the grid refinement study, second-order accuracy for the FDS method cannot be verified.

Method Comparison

To further compare the three methods, flow traveling through an oblique shock wave was studied. The oblique shock study is a more demanding and significant test than the source flow study. For the shock study, a uniform flow of Mach number 2.2 enters a channel, region 1, with initially parallel walls, see Figure 22. As the flow proceeds downstream, it encounters a ramp at the bottom wall with a deflection angle of 10 degrees. This ramp creates an incident oblique shock wave that turns the flow parallel to the ramp, forming a second area of uniform flow, which is represented by region 2 in the figure. Finally, the flow is turned back parallel to the wall by an oblique shock reflection. This results in uniform flow in region 3. All three methods were analyzed using 51 nodes on each computational plane. Table IV gives the exact conditions for the three regions:

Table IV. Exact values in oblique shock study.

Property	Region 1	Region 2	Region 3
static pressure (Pa)	206,842.0	344,829.8	548,689.3
static temperature (K)	1500.0	1666.8	1833.4
Mach number	2.2	1.885555	1.582704
specific heat ratio	1.25	1.25	1.25
gas constant (J/Kg/K)	332.56	332.56	332.56

Interior Point Method Comparison. A graph showing plots of all three methods computed across an oblique shock wave, along the streamwise grid location of j=41, can be seen in Figure 23. At first glance, it appears that all three methods predicted identical pressures for a given value of x. What this means is that if the pressures were calculated at every value of x, the values computed for all three methods would appear on the same curve. But, in a numerical scheme, values are only computed at discrete locations, which is determined by the step size. Both second-order interior methods compute the exact same value with identical step sizes, but the first-order interior method computes a different step size.

The reason that all three methods lie very near the same curve is because the magnitude of the second-order correction term is very small as compared to the first-order solution that it is added to. To explain why the second-order correction is small compared to the first-order solution, a discussion of the ideal Riemann problem for the oblique shock wave is in order. For an idealized Riemann problem, on each computational plane the shock wave is captured by two nodes and only passes through one Riemann location that is between the nodes. See Figure 24. In this case, the Riemann location above j, at j+1/2, contains nonzero values because it separates two different uniform flow regions, at j and j+1. Eq (38) can be used to demonstrate this:

$$(dE1)_{j+1/2} = [\ (dE1)_{wave\ 3} + (dE1)_{wave\ 2} + (dE1)_{wave\ 1}\ ]_{j+1/2} \quad \textbf{(38)}$$

As can be seen in Eq (38), which represents the total difference between nodes j and j+1, the flux differences, at j+1/2, would contain nonzero values. A similar argument can be made about the Riemann location that would be below j, at j-1/2. Because the uniform flowfields at j and j-1 are equal, there is no flux difference at j-1/2. Thus, the Riemann location at j-1/2, is in uniform flow and contains only zero values. The second-order correction computes the difference between the positive biased split flux differences above and below the node, and the differences between the negative biased split flux differences above and below. The differences of the positive and negative split flux differences from above and below would give values equal in magnitude to those at j+1/2 since the values of the split flux differences below are equal to zero. These differences of split flux differences are multiplied by a coefficient for the second-order corrections and are then added to the first-order solutions. In the idealized case, values on the order of the coefficients would be added. In the case where the shock is smeared over four or five nodes, the values at the Riemann locations are also smeared. This means that the values in two adjacent Riemann locations are very near one another. As a result, the differences of the positive and negative split flux differences above and below the node are very small, and when multiplied by the coefficients, they become much smaller in comparison with the first-order solution. Since the second-order corrections that are added to the first-order solution is much smaller than the first-order solution, all three methods appear to predict values along the same curve.

Boundary Point Method Comparison. A graph showing plots of all three methods computed on the upper wall across a shock reflection can be seen in Figure 25. Method11 and method21 show a monotonic convergence behavior across the shock. This is the expected result since both methods are first-order accurate on the boundary. As noted above in the interior point method comparison, these two methods appear to predict values that fall almost on the same curve with the only difference being step size. For method22, the solution moves in the wrong direction just before the shock, and slightly overshoots the exact solution after the shock. The initial drop in the value before the shock is caused by the linear extrapolation of the values that are located at j-1/2 and j-3/2. Recall that this linear extrapolation was needed to get values for split flux differences at a fictitious Riemann location just outside the boundary. Information was needed there to compute the differences of split flux differences for the second-order corrections. The problem arises because the Riemann location at j-3/2 may be in region 2, behind the shock wave, but j-1/2 may still be influenced by region 1, as illustrated in Figure 26. This results in erroneous information being extrapolated to the fictitious Riemann location. The same problem arises when the computational plane is just behind the shock reflection. Riemann location j-1/2 is influenced by region 3, but location j-3/2 is still in region 2. As a result, the second-order boundary correction looses accuracy around a shock reflection.

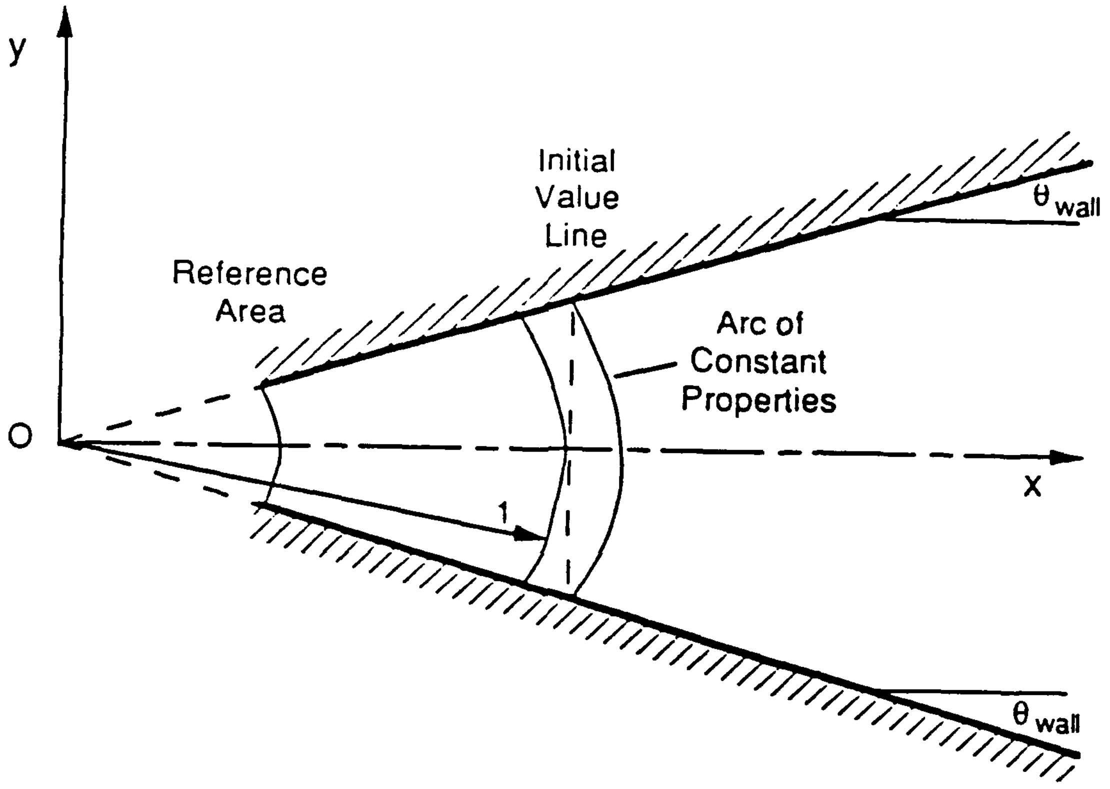


Figure 15. Geometry for planar supersonic source flow (3:237).

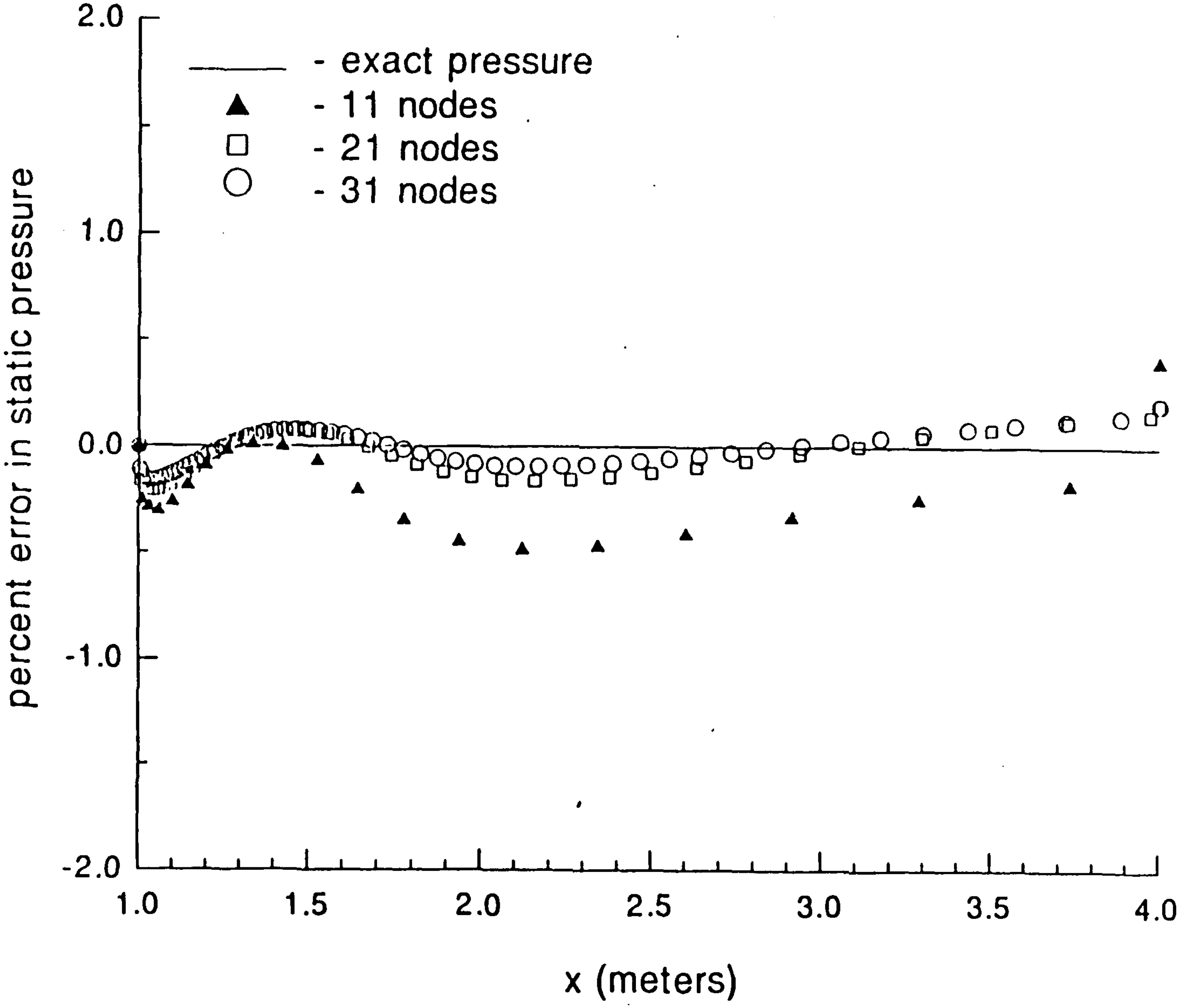


Figure 16. *Source flow centerline error comparison for method11.*

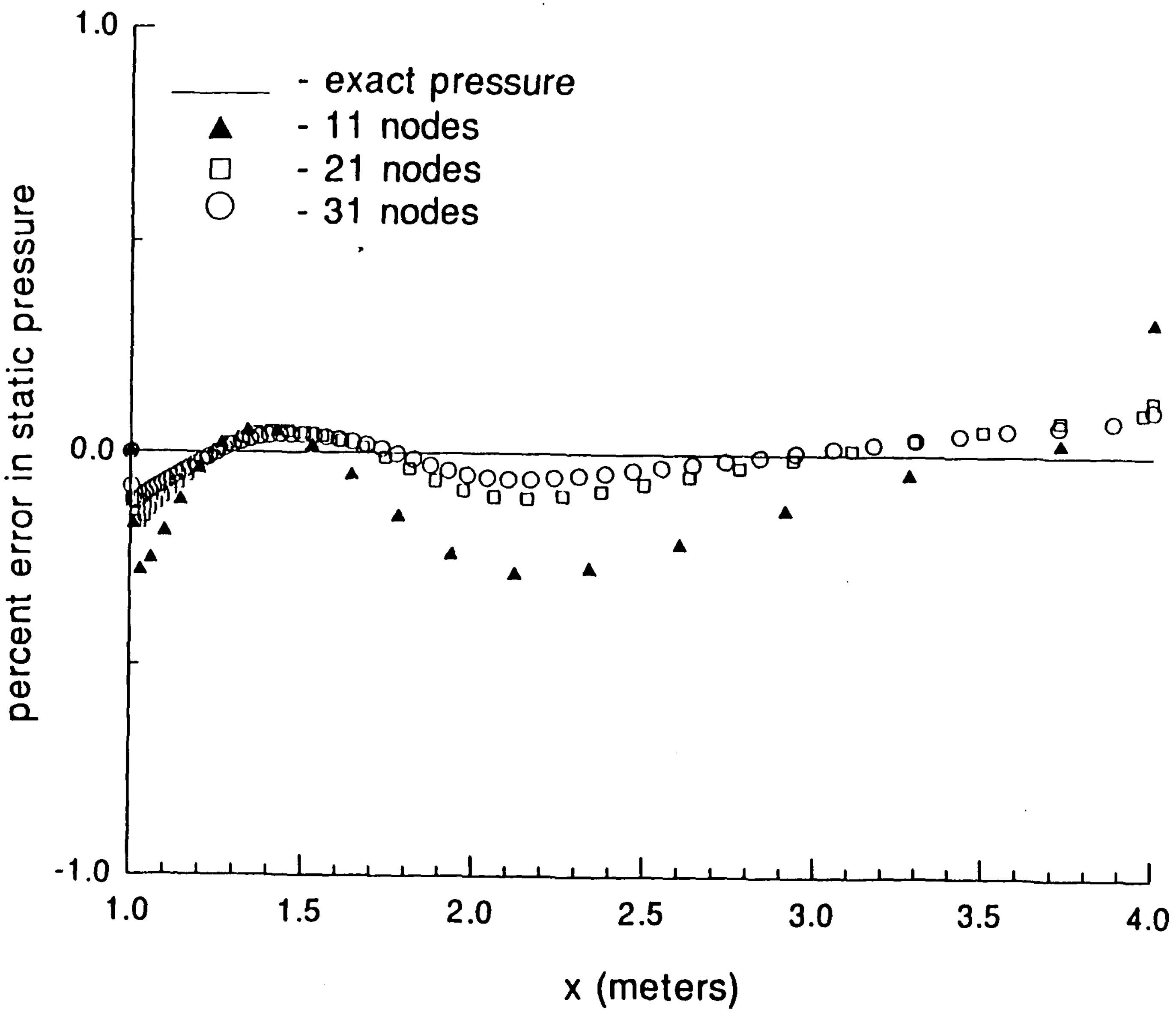


Figure 17. Source flow centerline error comparison for method21.

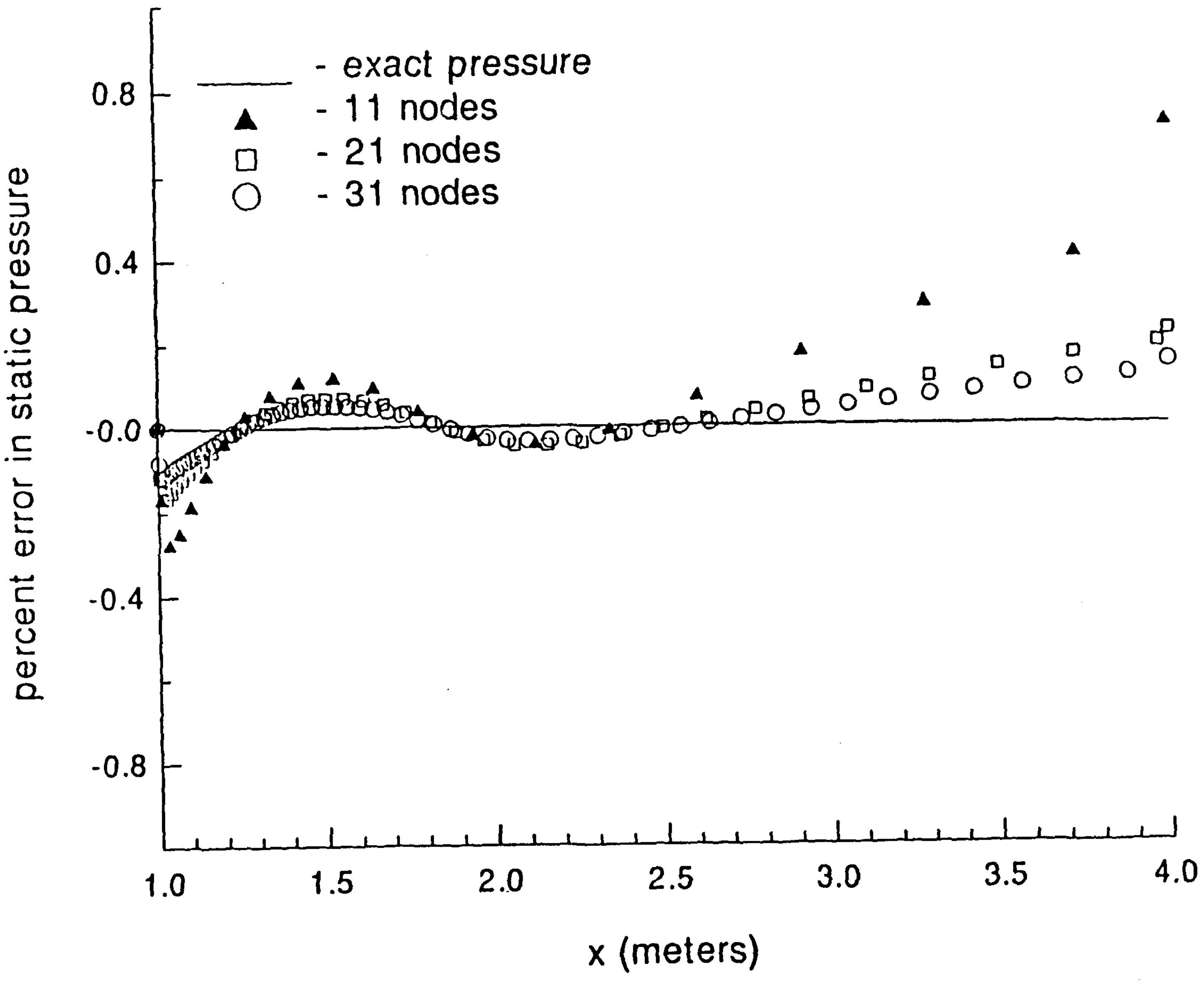


Figure 18. Source flow centerline error comparison for method22.

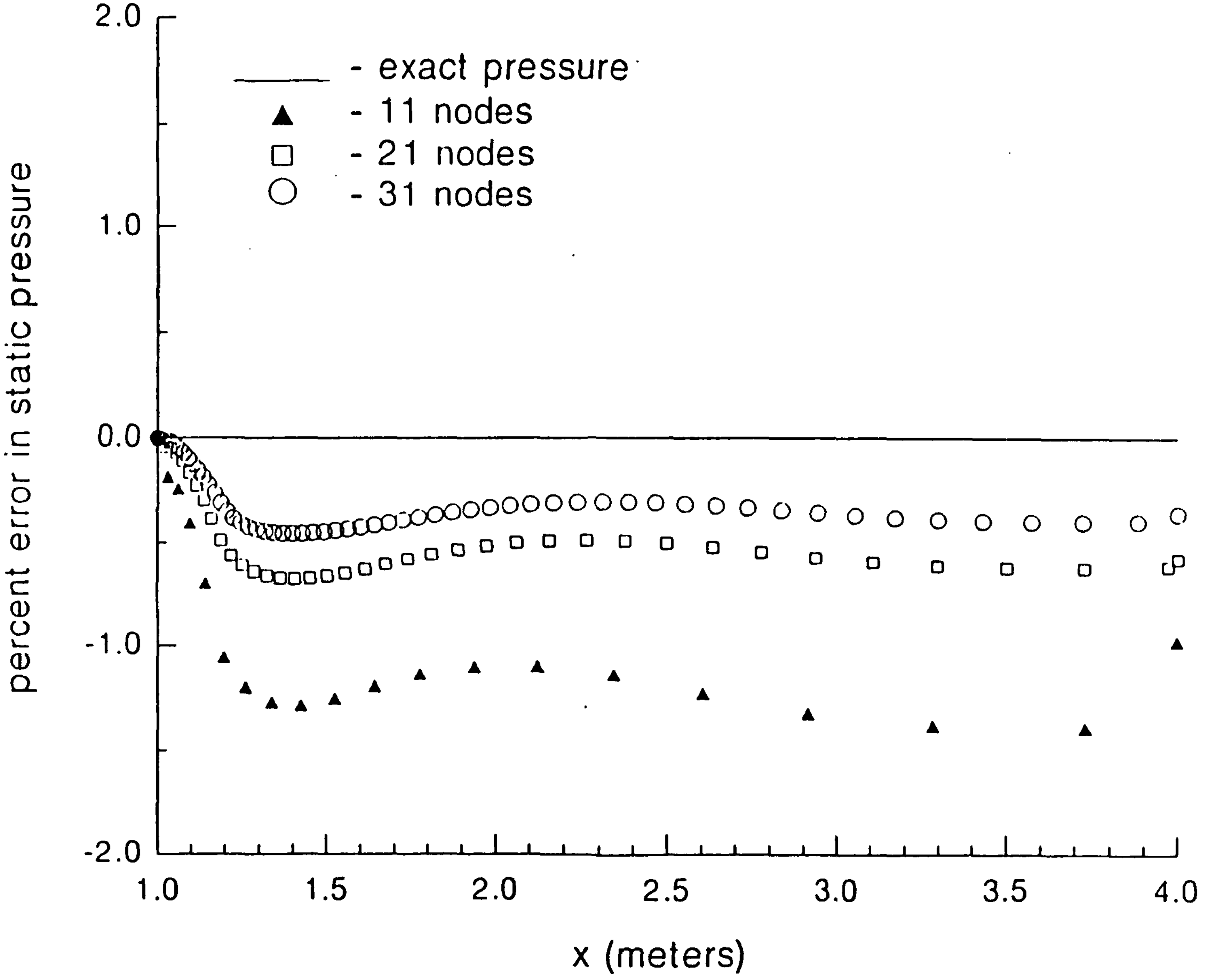


Figure 19. Source flow upper boundary error comparison for method11.

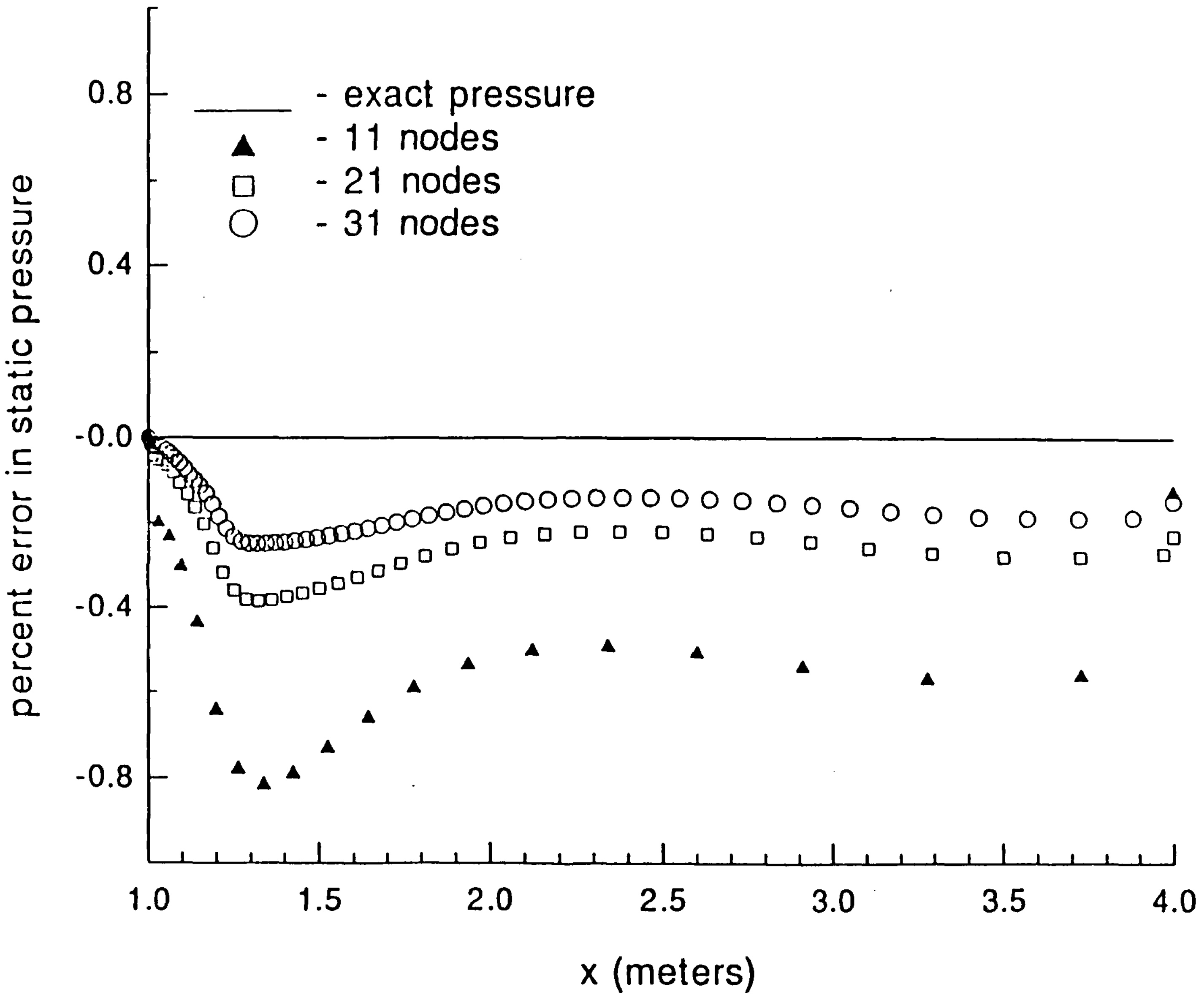


Figure 20. Source flow upper boundary error comparison for method21.

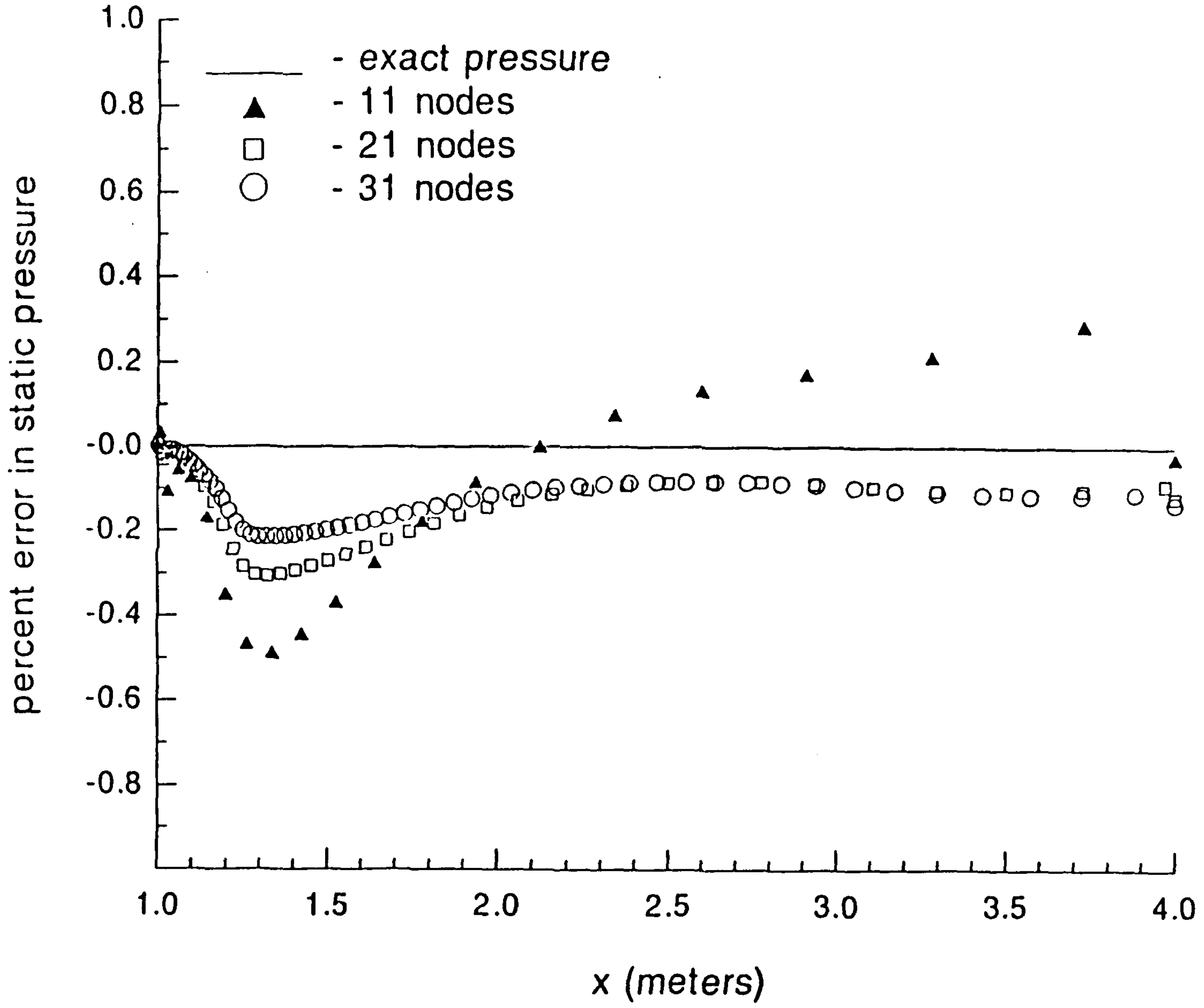


Figure 21. Source flow upper boundary error comparison for method22.

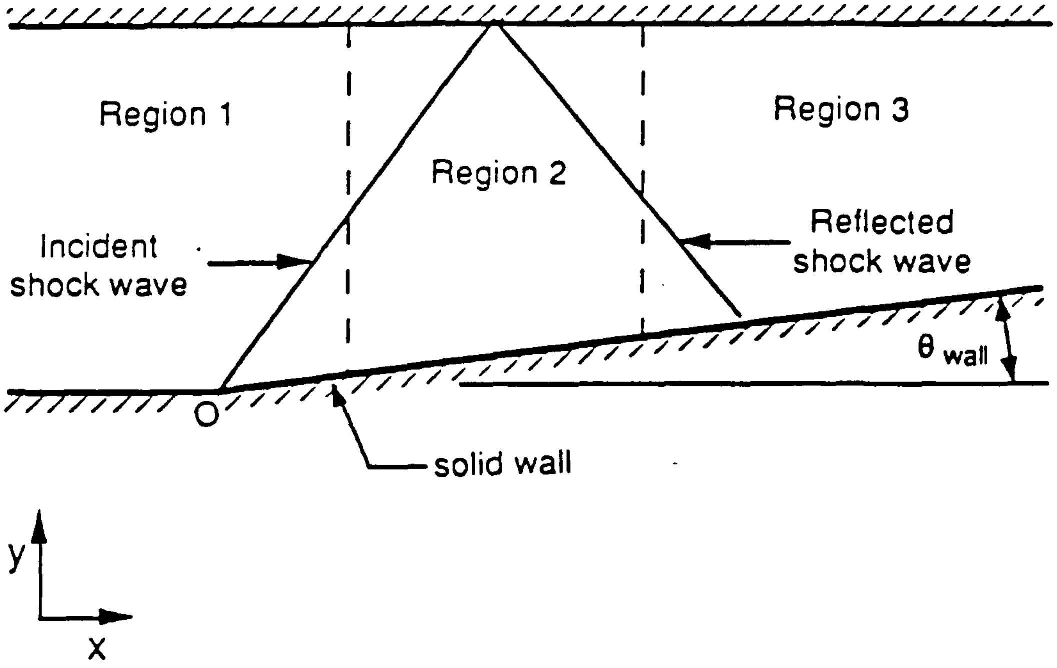


Figure 22. Geometry for shock wave reflection study (3:44).

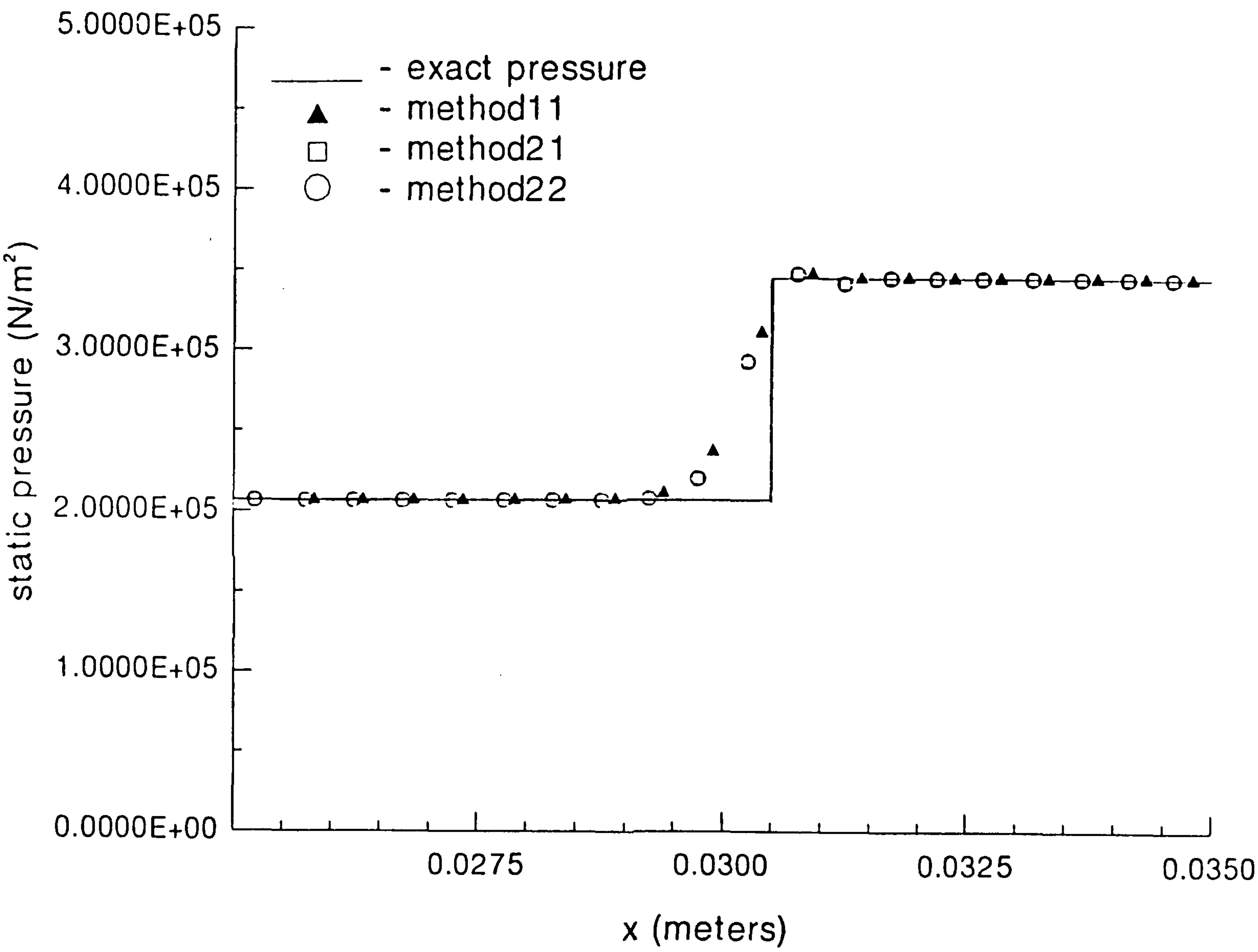


Figure 23. Method comparison for flow across an oblique shock wave at an interior point.

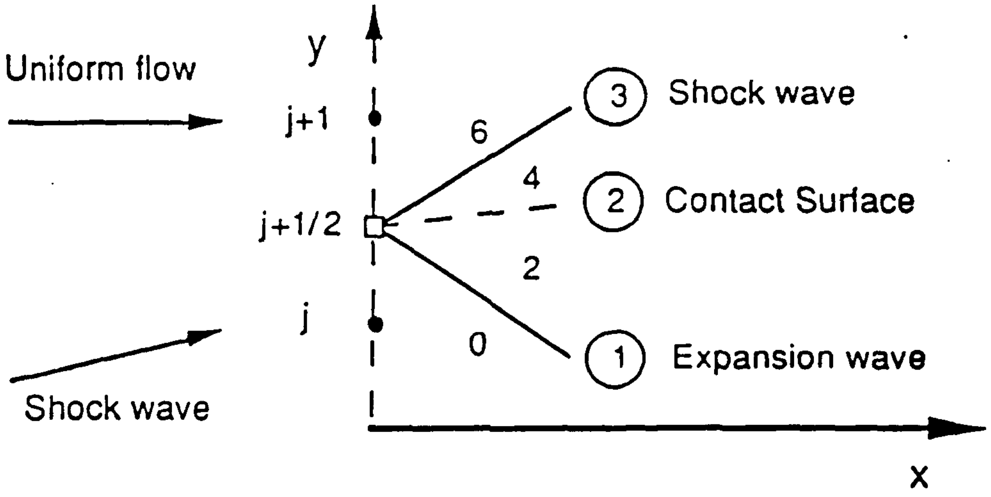


Figure 24. Riemann problem for a shock wave (3:180).

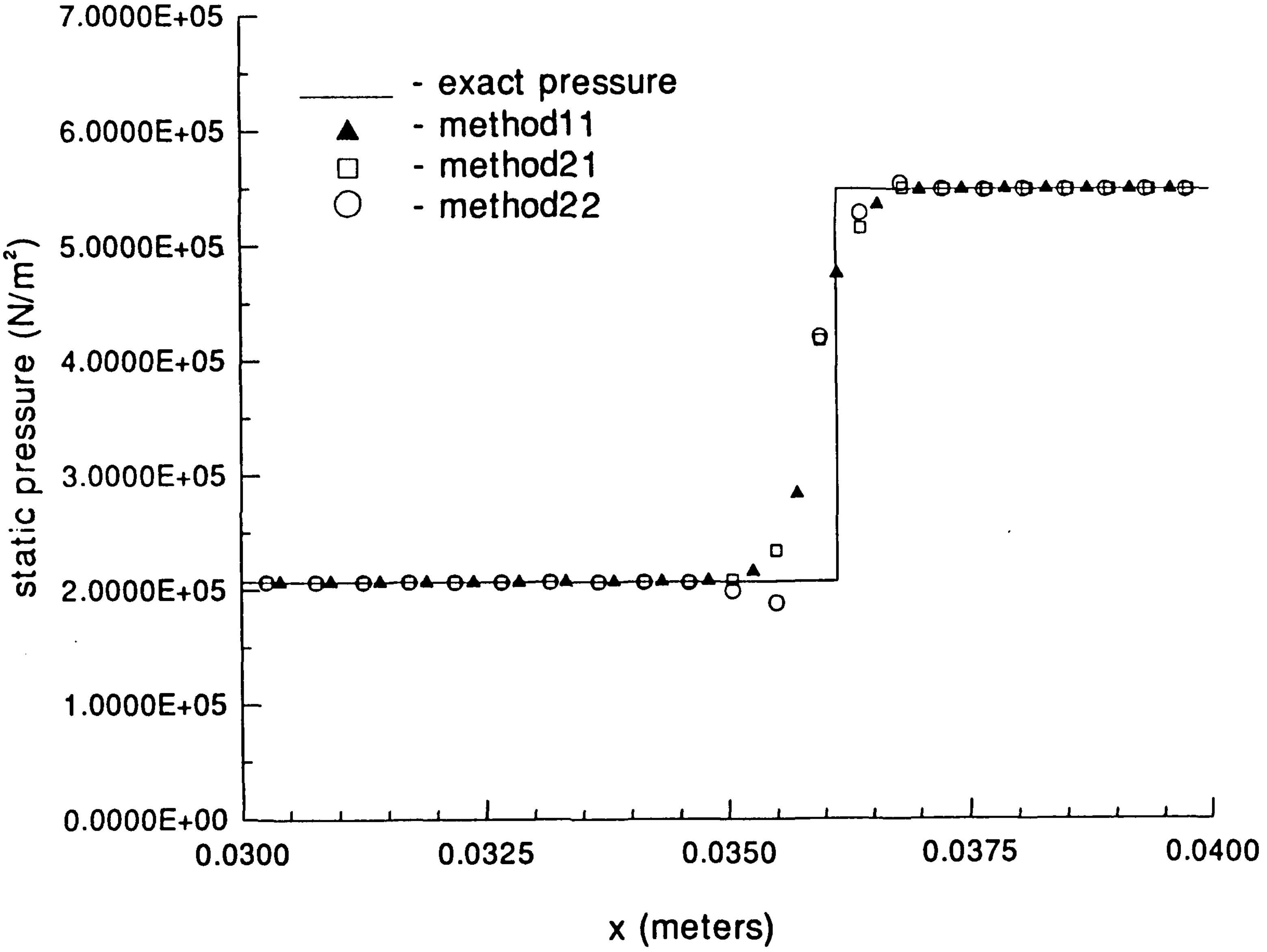


Figure 25. Method comparison for flow across a shock reflection on the upper boundary.

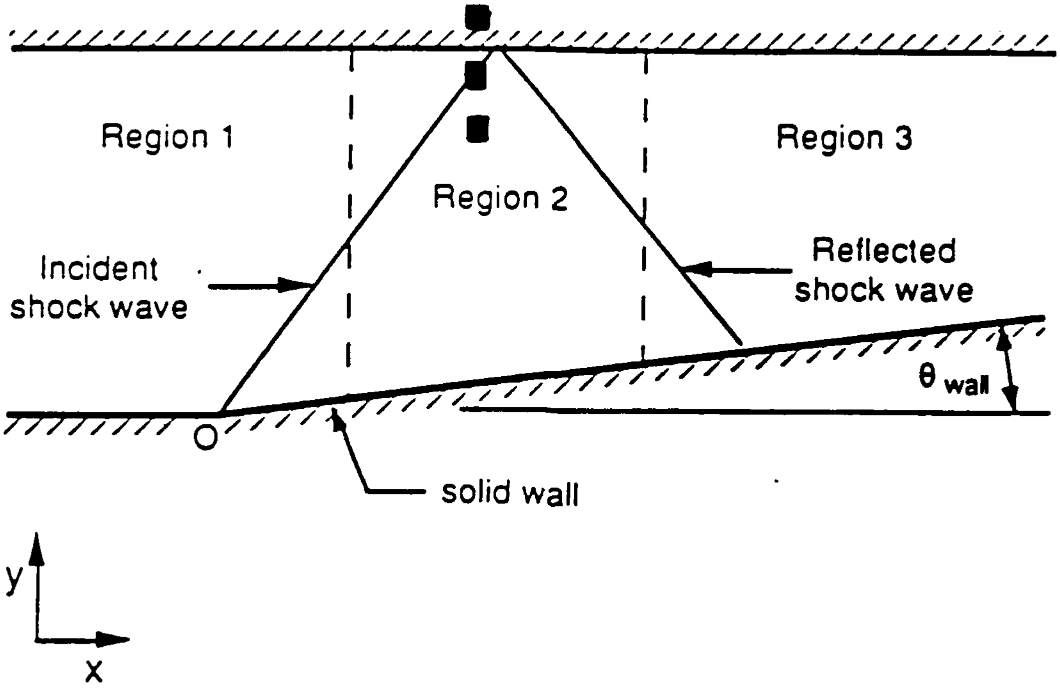


Figure 26. Linear extrapolation across a shock at a boundary (3:44).

VII. Summary

Second-order correction terms were added to an existing computer program that is based on a first-order accurate flux-difference-splitting method. Individual analyses were performed in an effort to verify the second-order accuracy of the solution and to determine the benefits associated with the second-order method.

A grid refinement study was performed in an attempt to verify a second-order accuracy in the solution obtained by adding the second-order correction terms to the first-order solution. Second-order accuracy could not be confirmed. Additionally, it was noted for all three methods evaluated that doubling the number of nodes resulted in an error reduction of approximately 50 percent. The only exception to this 50 percent reduction was seen in the method22 analysis at the boundary, which had a reduction of only 35 percent. This boundary behavior for method22 resulted from extrapolating information from two interior Riemann locations that were in different flow regions to a fictitious Riemann location outside the boundary, and then weighing the fictitious information equally with real information from the location just inside the boundary.

During the oblique shock reflection study it was observed that, as a result of shock smearing, the second-order correction for the interior points was very small in magnitude with respect to the first-order solution. Consequently, all three methods predicted values that appeared to lie on the same curve. The only deviation was that the step size for the second-order methods is different than that of the first-order method. The shock reflection study on the boundary demonstrated that method11 and method21 predicted similar values. This was expected since the two methods are first-order accurate on the boundary. Again, the only deviation was in the step size and the axial location of the computational planes. Method22 performed poorly on the boundary. The reason for the poor performance for method22 was because it uses information extrapolated from the first two interior Riemann locations to assign information to a fictitious Riemann location outside the boundary. *The problem lies in* the fact that the method extrapolates across the oblique shock wave, resulting in erroneous values at the nonphysical Riemann location. This anomaly occurs both in front of the shock reflection and behind it.

As a result of the above observations and conclusions, and with the knowledge that the first-order method has a faster computation time and is as accurate as many second-order finite differencing schemes, Taylor (8:108), the recommendation is to perform future analysis using the first-order method.

Bibliography

1. Anderson, Dale A., Tannehill, John C. and Richard H. Pletcher. Computational Fluid Mechanics and Heat Transfer. New York: Hemisphere Publishing Corporation, 1984.

2. Barthelemy, Robert R. "The National Aerospace Program: A Revolutionary Concept," APL Technical Review, Volume 2, Number 1, (1990).

3. Doty, John H. Maximum Thrust Planar Supersonic Nozzles Using A Flux-Difference Splitting Technique. Ph.D. Dissertation. Purdue University, Indiana. August 1991.

4. Doty, John H., Thompson, H. Doyle, and Joe D. Hoffman. Optimum Thrust Supersonic Nozzles for Hypersonic Vehicles, Volume 1: Theory and Results. Purdue University, Indiana. (To be published).

5. Enquist, B. and S. Osher. "Stable and Entropy Satisfying Approximations for Transonic Flow Calculations," Mathematics of Computation, Volume 34, (1981).

6. Hindman, Richard G. "Geometrically Induced Errors and Their Relationship to the Form of the Governing Equations and the Treatment of Generalized Mappings," AIAA Paper 81-1008, Palo Alto, California (1981).

7. Pandolfi, M. "A Contribution to the Numerical Prediction of Unsteady Flows," Journal of the American Institute of Aeronautics and Astronautics, Volume 22, Number 5. (1984).

8. Taylor, T. D., Ndefo, E., and B. S. Masson. "A Study of Numerical Methods for Solving Viscous an Inviscid Flow Problems," Journal of Computational Physics, Volume 9, Number 5, (1972).

9. Zucrow, Maurice J. and Joe D. Hoffman. Gas Dynamics, Volume 1. New York: John Wiley & Sons, 1976.

Appendix A: Weak Conservation Law Form of the Governing Equations

The weak conservation form uses metrics at the Riemann locations and is, therefore, consistent with the FDS. This translation of the metrics is a logical step because the split flux differences, which pass information to plane i+1, are computed at the Riemann locations. Converting the governing equations into weak conservation form is done by moving the metrics in the governing equations, Eq (10), into the derivative, and differencing the flux/metric product, Hindman (6:113). A mathematical manipulation is require to move the metrics into the differentiation that occurs in Eq (10), which is repeated here for convenience:

$$\frac{\partial(\boldsymbol{E})}{\partial\zeta} = -\eta_x\frac{\partial(\boldsymbol{E})}{\partial\eta} - \eta_y\frac{\partial(\boldsymbol{F})}{\partial\eta} \qquad \textbf{(10)}$$

Using the product rule of partial differentiation, it can be demonstrated that the metrics can be moved directly into the differentials on the right-hand side of Eq (10) as follows:

$$\frac{\partial}{\partial\eta}(\eta_x\,\boldsymbol{E}) = \eta_x\,\frac{\partial(\boldsymbol{E})}{\partial\eta} + \frac{\partial}{\partial\eta}(\eta_x)\,\boldsymbol{E} \qquad \textbf{(76)}$$

Manipulating the last term in Eq (76) shows

$$\frac{\partial}{\partial\eta}(\eta_x)\,\boldsymbol{E} = \eta_{x\eta}\,\boldsymbol{E} = \eta_{\eta x}\,\boldsymbol{E} = (1)_x\,\boldsymbol{E} = 0 \tag{77}$$

Substituting Eq (77) into Eq (76) and reversing the terms

$$\eta_x\,\frac{\partial(\boldsymbol{E})}{\partial\eta} = \frac{\partial}{\partial\eta}(\eta_x\,\boldsymbol{E}) \tag{78}$$

Equation (78) corresponds to the first term on the right-hand side of Eq (10). A similar manipulation can be performed on the last term in Eq (10). Therefore, it has been demonstrated that the metrics can be moved into the differentials while retaining the integrity of the governing equations. Thus, through a similar derivation of Eq (59), Eq (10) can be transformed into the first-order flux-difference-splitting, weak conservation law form:

$$\begin{aligned}\boldsymbol{E}_j^{i+1} = \boldsymbol{E}_j^{i} &- \Delta\zeta\left[(\eta_x\,d\boldsymbol{E}^{+})_{j-1/2} + (\eta_x\,d\boldsymbol{E}^{-})_{j+1/2}\right]\\ &- \Delta\zeta\left[(\eta_y\,d\boldsymbol{F}^{+})_{j-1/2} + (\eta_y\,d\boldsymbol{F}^{-})_{j+1/2}\right]\end{aligned} \tag{79}$$

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
December 1991

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE
INVESTIGATION OF A FLUX-DIFFERENCE-SPLITTING NUMERICAL METHOD IN SUPERSONIC NOZZLES

5. FUNDING NUMBERS

6. AUTHOR(S)
Danny P. Allen, Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GAE/ENY/91D-13

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This research investigated the effects of adding a second-order flux-difference-splitting (FDS) correction term to an existing computer code that is based on a first-order FDS algorithm. It was determined that the second-order algorithm did improve the accuracy of the code for a source flow analysis, but second-order behavior could not be confirmed by the error convergence patterns. It was also discovered that, when tested across an oblique shock wave, the second-order correction terms had minimal influence on the accuracy and shock capturing ability of the first-order accurate FDS method.

14. SUBJECT TERMS
Hypersonic Nozzle, Flux-Difference-Splitting, Riemann Solution, Second-Order Correction Terms, SCRAMJET

15. NUMBER OF PAGES
79

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT
Unclassified

20. LIMITATION OF ABSTRACT
UL